

**COST-BASED RISK PREDICTION AND
IDENTIFICATION OF PROJECT COST DRIVERS
USING ARTIFICIAL NEURAL NETWORKS**

Annie R. Pearce

Ph.D. Candidate

Construction Engineering and Management Division

School of Civil and Environmental Engineering

Georgia Institute of Technology

Atlanta, GA 30332-0355 USA

Final Report for:

Graduate Student Research Program

Tyndall AFB

Sponsored by:

Air Force Office of Scientific Research

Bolling Air Force Base, Washington, DC

And

Tyndall AFB

August 1997

COST-BASED RISK PREDICTION AND IDENTIFICATION OF PROJECT COST DRIVERS USING ARTIFICIAL NEURAL NETWORKS

Annie R. Pearce
Ph.D. Candidate
Construction Engineering and Management Division
School of Civil and Environmental Engineering
Georgia Institute of Technology

ABSTRACT

This research investigated the effectiveness of using Artificial Neural Networks (ANNs) to predict risks related to final project costs and to identify potentially significant cost drivers relating to construction projects. Several ANN models were developed which related potential project costs to a variety of input factors typically used to perform conceptual cost estimating. The models were deployed over a set of permutations of all input variables and used to generate a maximum cost vs. probability curve which can be used to evaluate risks of cost growth between conceptual design and project completion. Using the intrinsic feature representation properties of ANNs, cost drivers were identified which should be managed for projects to reduce the risks of cost growth. Results of the research indicate that ANNs can serve as a robust tool for cost estimation and approximated multivariate regression analysis.

COST-BASED RISK PREDICTION AND IDENTIFICATION OF PROJECT COST DRIVERS USING ARTIFICIAL NEURAL NETWORKS

Annie R. Pearce

INTRODUCTION

As budgets for the construction of new facilities and the maintenance, decommissioning, or rehabilitation of existing facilities become more limited, stakeholders responsible for funding these activities are seeking better ways to increase the accuracy of their project cost estimates, especially in the early phases of project planning and design. Being able to predict how much a project will cost is important not only to set sufficient funds aside to complete a project, but also to ensure that the project is not over-budgeted, potentially encouraging cost overruns or preventing other deserving projects from receiving adequate funding.

Alternative methods for cost prediction are especially important in the early planning or conceptual design phases of projects, before enough detail is known to allow a traditional quantity takeoff estimate to be performed. Since much of the budgeting process often takes place during these early phases of a project, an accurate estimate is essential to ensure that projects are allocated a sufficient budget so that functional requirements are met. At the same time, an accurate project estimate can help to avoid over-budgeting that may encourage gold-plating during design, or remove efficiency and performance constraints during construction.

Accurate project cost predictions or estimates early in the planning and design processes can also serve as a cost-control measure to assist in managing the design process. With an understanding of the most significant factors affecting final project cost, i.e., cost drivers, project owners and managers can proactively make cost-effective choices during design, rather than after construction begins and budgeted dollars begin to fall short of requirements.

In this paper, we examine the potential of Artificial Neural Networks (ANNs) as a tool to support the tasks of cost prediction, cost driver identification, and risk management during the planning and design phases of the project life cycle. ANNs are a modeling tool based

loosely on the computational paradigm of the human brain, and have proven to be robust and reliable for tasks of prediction, ranking, classification, and interpretation or processing of data. We begin by examining the problem of project cost prediction in more detail, followed by the objectives of the research and a description of the methodology followed, including some background on the theory and implementation of ANNs. The results of the research and a discussion of their implications form the primary contribution of this work. The paper concludes with a look at future research and applications that can stem from this proof of concept.

PROBLEM STATEMENT

As a point of departure for the research described in this report, we first needed to establish the nature, parameters, and objectives of the problem of project cost management before developing a strategy for approaching the research. The following sections describe the background to the problem and objectives guiding the research.

Background to the Problem of Project Cost Management

The problem of managing project costs is not new. The whole procurement paradigm of Project Management was created as a response to the need to ensure that projects are completed on time and within budget, to an acceptable standard of quality (Figure 1). However, this paradigm is typically most influential during the construction phase of a project life, whereas the greatest impact with the least effort can be had on final project cost much earlier in the life cycle, namely in the planning and design phases (Figure 2). In fact, most of the critical decisions affecting the total cost of a project are made during the project planning phase, before designers, project managers, and contractors typically join the project team (Burns 1997).

In today's cost-conscious project environment, project owners and planners need a way to predict how their early decisions will ultimately impact the final cost of a project. While this need has traditionally been addressed by heuristic or "rule-of-thumb" knowledge (e.g., "the larger the building perimeter, the greater the cost of exterior enclosure"), no quantitative method currently exists for understanding how planning choices affect final project costs.

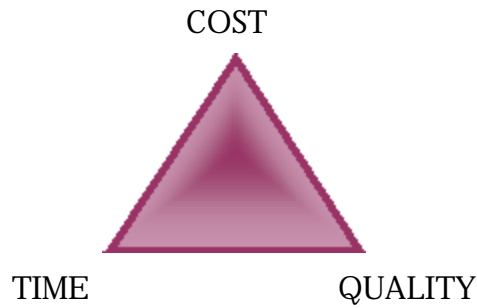


Figure 1: Traditional Concerns in Managing a Construction Project

This lack of quantitative method is due primarily to the complexity of analyzing the multiple factors influencing final cost. With the multitude of interacting variables that potentially affect cost even in the early planning stages of projects, performing a rigorous multivariate nonlinear regression to determine the relative importance of those variables is a nontrivial computational task.

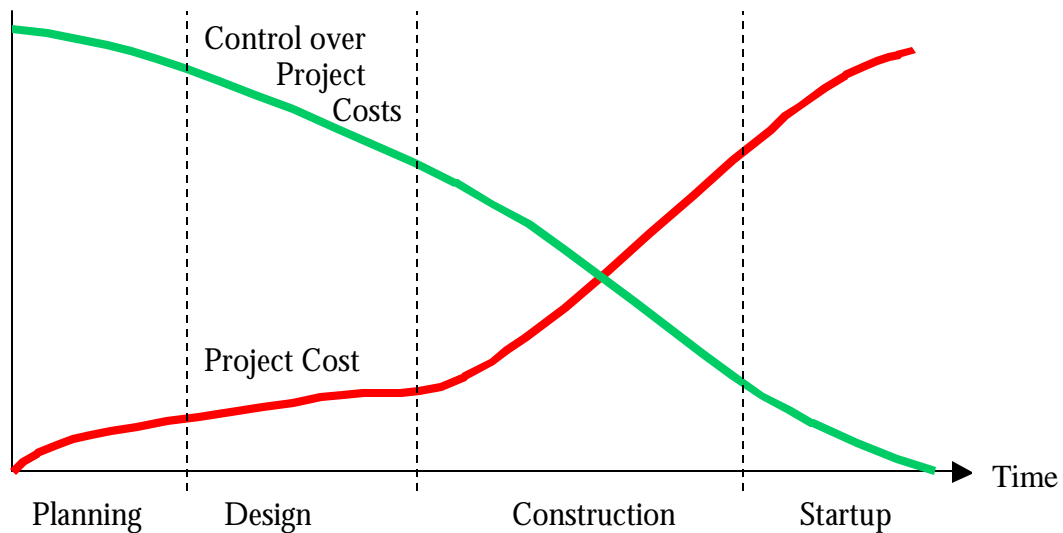


Figure 2: Cost and Control vs. Time for a Construction Project

Artificial Neural Networks (ANNs) offer an alternative to traditional methods of cost prediction based on parametric or quantity takeoff techniques. With their capacity to learn from examples and to generalize that knowledge to novel cases, ANNs provide the ability to undertake rapid modeling of systems in which the interaction between input and output

variables is unknown but where representative examples of inputs and outputs exist. ANN modeling of the process of project cost prediction potentially provides important clues to the relationships between initial planning-phase project variables and final cost. ANN models were used in this research as a quantitative approach to identifying cost drivers and risk factors that can be used to manage project planning and design.

Research Objectives

The objectives of this research were twofold:

- To develop a quantitative methodology for identifying and ranking significant project cost drivers.
- To develop a prototype cost prediction model useful for generating range estimates of final project costs with limited knowledge of project details.

These objectives were addressed to meet the needs of project owners during the planning phase of construction projects for guidance on which factors should be most closely managed to result in projects that meet functional requirements while remaining within budget. The range estimating capability of the model enables project planners to identify the potential for cost variation by the end of the project, thus facilitating the budgeting process.

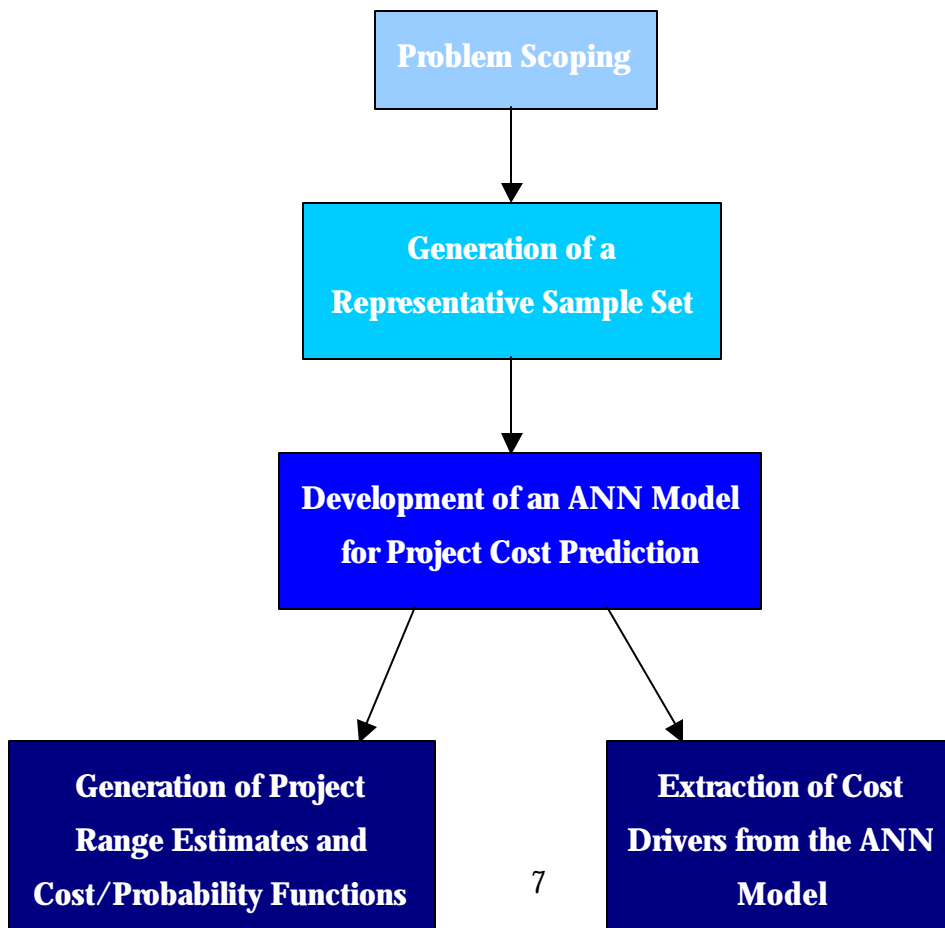
METHODOLOGY AND RESULTS

The methodology undertaken to achieve the project objectives is presented in Figure 3. The following sections describe each of the primary phases of the research, along with the key results obtained at each phase.

Research Scoping

The first phase of the research involved scoping the work to narrow the population of construction projects to be considered in developing the ANN model. In order to demonstrate proof of concept, we sought to minimize the number of input variables considered, while still maintaining a sufficient degree of problem complexity to keep the relationships between inputs and outputs unpredictable using traditional regression techniques. Scoping was also based on the availability of simulation tools to generate scenarios or examples to be used to train the ANN model.

Figure 3: Research Methodology



We elected to limit our attention to vertical construction, since we had the Parametric Automated Cost Engineering System (PACES) model to use for scenario simulation. This cost estimating tool provides the capability to model total project costs for a variety of typical commercial building types, ranging from communication centers to medical facilities to living quarters (Williams 1997). Within each category of building types in PACES, various models are included to estimate costs for specific building configurations and functional capabilities. We selected a specific facility type and functional model based on the availability of an existing database of real project data to be used for validation. The type of facility we selected was dormitories, and the specific model we chose to simulate was for enlisted personnel dormitories in a "1+1" configuration, i.e., suites of two rooms, each housing one enlisted person, with shared bath and living areas. This type of dormitory comprises the majority of new military enlisted dormitory construction (Burns 1997), and thus has strong potential for initial research results to have an impact on future construction practice. The military enlisted "1+1" dormitory model provided in PACES is comparable to many dormitory designs also used in civilian commercial construction, such as the dormitories constructed for university campuses.

To further scope the problem, we chose to investigate various facility configurations for a constant number of inhabitants. In the PACES model, number of inhabitants drives the total floor area of the facility. By keeping the number of inhabitants constant, we limited the scope of variability to two main classes of variables (geometric variables and architectural variables), while keeping factors constant that are typically fixed early in the planning process. The parameters that were fixed over the sample set are shown in Table 1.

Table 1: Parameters Fixed for Problem Scoping

Parameter	Value	Comments
Facility Type	Dormitory	Selected due to availability of real cases for validation.
Building Model	Enlisted “1+1”	Greatest number of empirical cases; generalizability to civilian construction.
Number of Persons	156	Fixed to maintain constant building square footage; typically known early in the planning phase. This is the default number of persons for the PACES Dormitory model.
Location	Atlanta, GA	The index city for the cost database. Fixed since location is typically known early in the planning phase. Affects seismic and foundation conditions, and heating and cooling loads.

Generation of a Representative Sample Set

After the parameters discussed in the previous section were fixed, the next task of the research was to generate a set of samples by varying the remaining input variables in a way that is representative of the range of possible design and construction practices. All data sets for this research were simulated using the PACES cost estimating tool. Simulated data was selected for model development instead of real data, due to the field-demonstrated accuracy of the PACES model (Burns 1997).

Despite the relative savings in labor due to using simulated data instead of empirical data, developing cases using PACES is still quite labor-intensive. Thus, finding a modeling technique that works with representative rather than comprehensive data sets was an important consideration in selecting ANNs to model project costs. Due to the massively interconnected and parallel nature of ANNs, a well-trained ANN model has the ability to generalize, i.e., provide reasonable outputs given a set of inputs to which it has not previously been exposed. This quality implies inversely that ANN models can be trained using sample sets which are not comprehensive but are instead representative, thus saving significant time and effort in simulating training cases.

Determining what cases should be simulated to generate a representative sample set was itself a nontrivial task, given the relatively large number of input variables, the large number

of potential values for each variable, and the nonlinear interactions between input configurations. For this initial proof of concept, only selected PACES input variables were manipulated, namely those which were hypothesized to have the most influence on final project cost. Table 2 shows selected input variables that could be manipulated to generate the sample set used in this research, along with possible values for each variable. While many of the input variables in isolation had a linear relationship to total project cost, in combination the variables interacted to result in a complex nonlinear cost function. Thus, significant effort was required to determine what values should be used for each variable to generate a representative sample set, while at the same time minimizing the number of scenarios that had to be generated using the PACES model. Appendix A describes in more detail the methodology used to select the scenarios comprising the sample set, along with the interim results of PACES experimentation.

As can be seen by inspecting Table 2, the number of possible permutations in a comprehensive data set is quite large, especially depending on the increments of sampling for continuous variables such as building perimeter or floor to floor height. Thus, determining a sampling strategy that would minimize the number of cases to be simulated was an extremely important task. The final set of parameters used to generate the representative sample set for model development is shown in Table 3.

Table 2: Selected PACES Parameters and Possible Values

Parameter	Possible Values
Stories Above Grade	0 – 10
Perimeter	50 – 5,000 Sq. Ft.
Floor to Floor Height	0 – 50 Ft.
Floor to Ceiling Height	0 – 50 Ft.
Soil Bearing Capacity	Low, Average, High
Floor Structure Type	Concrete Frame Steel Frame with Reinforced Concrete Deck Steel Frame with Metal Joists/Steel Deck/Concrete Fill Load Bearing Walls with Metal Joists/Steel Deck/Concrete Fill Load Bearing Walls with Wood Joists/Wood Deck Load Bearing Walls with Precast/Prestressed Floors
Bay Size/Span Length	Small, Average, Large
Roofing Type	Single Membrane Built-Up Shingle Standing Seam Metal Clay Tile Metal (Typical Metal Building)
Exterior Wall Type	Brick Veneer 4” Split Rib Masonry Veneer 8” Split Rib Masonry 8” Masonry Block Tilt-up Concrete Exposed Aggregate Precast 12” CIP Concrete with Exposed Aggregate Finish Metal Sandwich Panel Stucco E.I.F.S. (Dryvit)

Varying the input parameters shown in Table 3 resulted in a total of 46 training cases, as indicated by X's in the matrix. Five additional randomly-selected cases were developed to serve as test cases for assessing the model's performance with novel inputs. These cases are indicated by Ts in Table 3.

Table 3: Parameter Values Used in Sample Set

Floors Above Grade		1	2	3	4	5	6	7	8	9	10
Perimeter	Default	X	X	X	X	X	X	X	X	X	X
	2000 LF	X	X	X							
	3000 LF	X	X	X			T				
	4000 LF	X	X	X							
Floor to floor height	8 FT	X	X	X	X		X		X		
	10 FT			X				T			
	12 FT	X	X	X	X		X		X		
	14 FT	T		X							
	16 FT	X	X	X	X		X		X		T
	18 FT			X		T					
	20 FT	X	X	X	X		X		X		

Development of an ANN Model for Project Cost Prediction

One of the first tasks in developing an ANN model was to determine an acceptable threshold for error in output. An ANN model can be manipulated in many ways to improve its performance, including varying its internal architecture, learning paradigm or parameters, or modifying the data set used to “train” it. Given the large number of possible network configurations, selecting an acceptable level of error is important to scoping the process of network experimentation. Upon developing a network that performs within an acceptable level of error, further experimentation to improve accuracy is unnecessary. To select a threshold of acceptable error for the cost estimation problem, we used a range of acceptability of $\{+25\%, -10\%\}$. The upper limit was based on the cost variation authorized by the U.S. Congress for military construction projects (USC 1995), while the lower limit was based on standard industry practice for vertical construction projects (Rast 1997, Gregory 1997). Thus, an ANN model which could predict direct project costs within $\{+25\%, -10\%\}$ was considered acceptable for the purposes of this research.

The next steps in developing an ANN model to predict project costs were to select a network paradigm, and to transform the data from the PACES simulations into a form that could be fed to the network. The back-propagation paradigm of ANNs was selected, due to its demonstrated accuracy in problems of prediction and the transparency of logic underlying the theory of back-propagation neural networks. Additional detail about the back-

propagation class of ANN models is included in Appendix B of this report. Transformation of data from the PACES simulations involved “squashing” each value for input and output variables to lie between {0, 1}. Each value was squashed using a linear compression formula:

$$X_{\text{squashed}} = (X_{\text{original}} - X_{\text{min}}) / (X_{\text{max}} - X_{\text{min}})$$

where X_{squashed} = squashed value for variable X
 X_{original} = original value for variable X
 X_{min} = minimum value over all instances of variable X
 X_{max} = maximum value over all instances of variable X

In addition to the input variables shown in Table 3, the PACES simulator automatically varied other quantitative parameters based on initial input values. The total set of twelve variables used as input to the model is shown in Table 4.

Table 4: Input Variables for the ANN Model

Input Variable	Unit	Minimum Value	Maximum Value
Floors Above Grade	EA	1	10
Perimeter	LF	604	4000
Footprint	SF	6313	62316
Roof Area	SF	6682	65961
Floor to Floor Height	LF	8	20
Exterior Wall Area	SF	16308	129744
Exterior Window Area	SF	1201	9557
Exterior Doors	EA	4	41
Number of Stairwells	EA	0	4
Number of Elevators	EA	0	2
Heating Load	MBH	649	1137
Cooling Load	Tons	77.62	123.91

Two different configurations of outputs were considered: a single total direct cost output, and a direct cost split by Unifomat categories. The second configuration was selected, resulting in an initial set of data with 15 outputs per case – one for each Unifomat category (see Table 5). Eight of these initial variables were excluded from the transformed data, since they did not vary as a result of manipulating input variables. The seven output variables used to train the network are indicated in Table 4.

Table 5: Output Variables for the ANN Model

Uniformat Category	Used in Model?
Substructure	yes
Superstructure	yes
Exterior Closure	yes
Roofing	yes
Interior Construction	yes
Interior Finishes	no
Conveying Systems	yes
Plumbing	no
H.V.A.C.	yes
Fire Protection Systems	no
Electric Power and Light	no
Electrical Systems	no
Equipment	no
Furnishings	no
Special Construction	no

After configuring the 46 training cases to serve as network input, the next step was to begin experimentation with back-propagation ANN models to obtain the required level of accuracy in predicting project direct costs. The aforementioned error thresholds were used to evaluate network performance by summing the outputs of all Uniformat subsystems and comparing the total to the PACES predicted total cost for these subsystems.

Despite many advances in the theory of ANNs, choosing an appropriate network paradigm and architecture is still largely art rather than science. Various configurations of numbers of processing elements and hidden layers were tried in the general class of back-propagation ANNs. After initial experimentation, the Delta learning rule was selected to govern the ANN training, since it seemed to provide the best performance. Back-propagation is one paradigm for network learning that involves changing connection weights between hidden units based on the contribution each has made toward generating an erroneous output during training (see Wasserman 1989, NeuralWorks 1996 for more explanation). In addition to experimentation by trial and error with various network architectures, the cascade correlation algorithm (Feldman & Lebiere 1993) for developing ANN architectures was used to guide the trial and error, using the Predict expert system ANN shell by NeuralWorks.

Various ANN architectures were tried, ranging from one hidden layer consisting of 0 processing elements to three hidden layers consisting of a total of 21 processing elements. The tested architectures, along with various error metrics, are shown in Table 6 and illustrated graphically in Figure 4.

Table 6: Network Architectures and Errors

Network Architecture (input-hidden-output)	Maximum % Error (% of predicted total cost)	Average Error (% of predicted total cost)
12-0-7	-10.31%	-5.21%
12-1-7	-13.72%	-6.18%
12-2-7	-10.60%	-5.31%
12-3-7	-7.16%	-4.15%
12-5-7	-7.22%	-3.79%
12-7-7	-8.67%	-4.11%
12-9-7	-10.56%	-4.90%
12-1-1-7	-13.74%	-6.21%
12-3-3-7	-9.19%	-4.27%
12-5-5-7	-11.26%	-4.94%
12-7-7-7	-10.59%	-5.17%
12-1-1-1-7	-10.19%	-5.17%
12-3-3-3-7	-10.55%	-5.49%
12-5-5-5-7	-9.41%	-4.20%
12-7-7-7-7	-10.86%	-5.18%
12-5-3-1-7	-7.28%	-3.96%
12-7-5-3-7	-9.78%	-4.43%
12-1-3-5-7	-9.13%	-4.20%
12-3-5-7-7	-11.00%	-4.85%

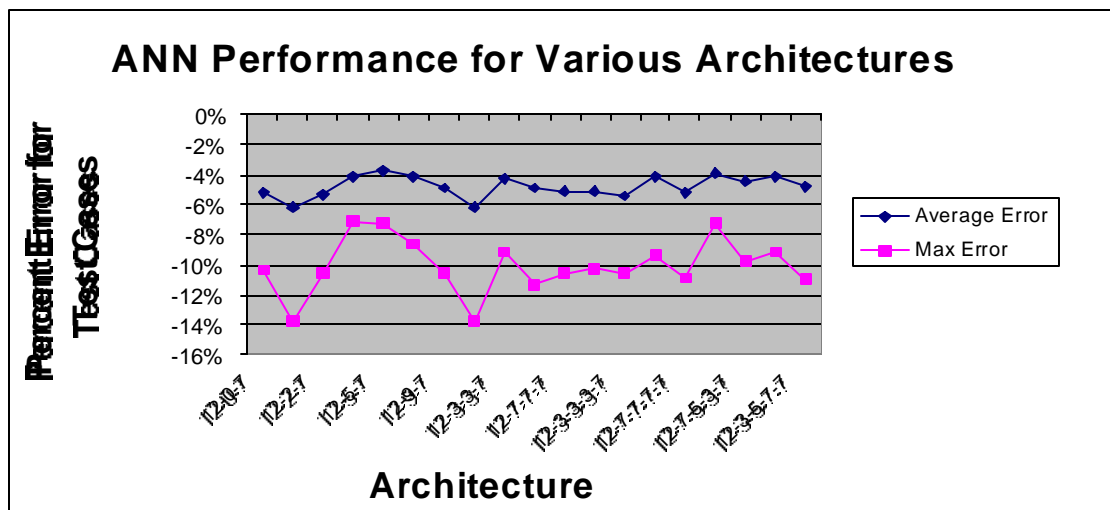


Figure 4: ANN Architectures vs. Error

Networks were trained for 50,000 passes of the training set, and used a learning coefficient of 0.3 with a sigmoid transfer function. The best performing network had an architecture of 12 input units, 7 output units, and one hidden layer with five processing elements. This network had an average error of -3.79% and a maximum error of -7.22% of in predicting total direct project costs over the test set.

While all ANN models seemed inclined to underestimate total direct costs on average, in fact many of the test cases resulted in slight overestimation by the models. The negative average error resulted in many cases from the fact that the models had difficulty in extrapolating to outlying cases, of which there was one in the test set. Despite this difficulty, even the worst-performing network had a maximum error of -13.74% , which is not far from the maximum industry-accepted threshold of -10% . Additional training and manipulation of ANN parameters is likely to improve even the worst-performing network to acceptable levels of performance.

Generation of Project Range Estimates and Cost/Probability Functions Using the ANN Model

After an acceptably trained ANN model was developed, the next step was to use the model to generate cost/probability functions over a comprehensive set of input variable permutations. While the PACES model or some other cost estimating tool could be used to perform this task, the lack of an automated scenario generation capacity made impossible the task of running a comprehensive set of input variables to generate corresponding outputs. Given an acceptably performing ANN model with batch capabilities, running a large set of cases was a simple computational task.

The ultimate outcome of this phase of the research was a cost/probability function to illustrate the potential range of project costs over all permutations of the input variables. After the shape of the cost/probability function is known, variations in final cost can be predicted based on the shape of the curve and the desired level of confidence required for the cost estimate. To generate the cost/probability function, a comprehensive set of input permutations was generated, with resolution of input variables increasing about their expected value as delineated in the PACES model. Resolution of input variables was

increased over three independent variables: number of stories, building perimeter, and floor-to-floor height. By increasing the resolution of these variables, a larger number of cases was generated about the expected value of the variables, resulting in a distribution more closely resembling the existing population of dormitories. Table 7 shows the input values used to generate the permutations comprising the comprehensive data set. Expected values for each parameter are indicated by a double-bar outline. The remaining input variables (footprint, roof area, exterior wall area, exterior window area, exterior doors, number of stairwells, number of elevators, heating load, and cooling load) were calculated using the PACES equations that depend on the three independent parameters. These equations can be found in Appendix C.

Table 7: Values for Independent Variables Used to Generate Comprehensive Data Set

Floors Above Grade		1	2	3	4	5	6	8	10
Perimeter	Default	X	X	X	X	X	X	X	X
	1000 LF	X	X	X	X	X	X	X	X
	1250 LF	X	X	X	X	X	X	X	X
	1500 LF	X	X	X	X	X	X	X	X
	2000 LF	X	X	X	X	X	X	X	X
	3000 LF	X	X	X	X	X	X	X	X
	4000 LF	X	X	X	X	X	X	X	X
Floor to floor height	8 FT	X	X	X	X	X	X	X	X
	8.5 FT	X	X	X	X	X	X	X	X
	9 FT	X	X	X	X	X	X	X	X
	9.5 FT	X	X	X	X	X	X	X	X
	10 FT	X	X	X	X	X	X	X	X
	10.5 FT	X	X	X	X	X	X	X	X
	11 FT	X	X	X	X	X	X	X	X
	12 FT	X	X	X	X	X	X	X	X
	14 FT	X	X	X	X	X	X	X	X
	16 FT	X	X	X	X	X	X	X	X
	18 FT	X	X	X	X	X	X	X	X
	20 FT	X	X	X	X	X	X	X	X

After the comprehensive data set was constructed and squashed (see **Development of an ANN Model.**), the best-performing ANN model was used to generate output values for each of the comprehensive data set cases. The resulting outputs were unsquashed and plotted as a histogram to generate an approximation of a cost/probability curve. The

histogram is illustrated in Figure 5. A cumulative frequency distribution of the ANN outputs is shown in Figure 6.

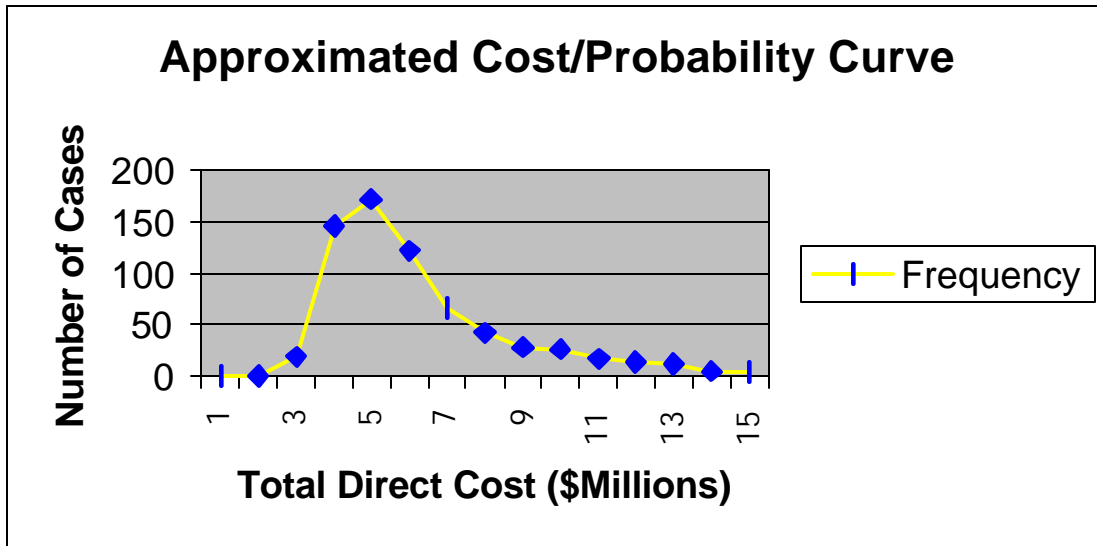


Figure 5: Frequency Distribution of Total Project Costs

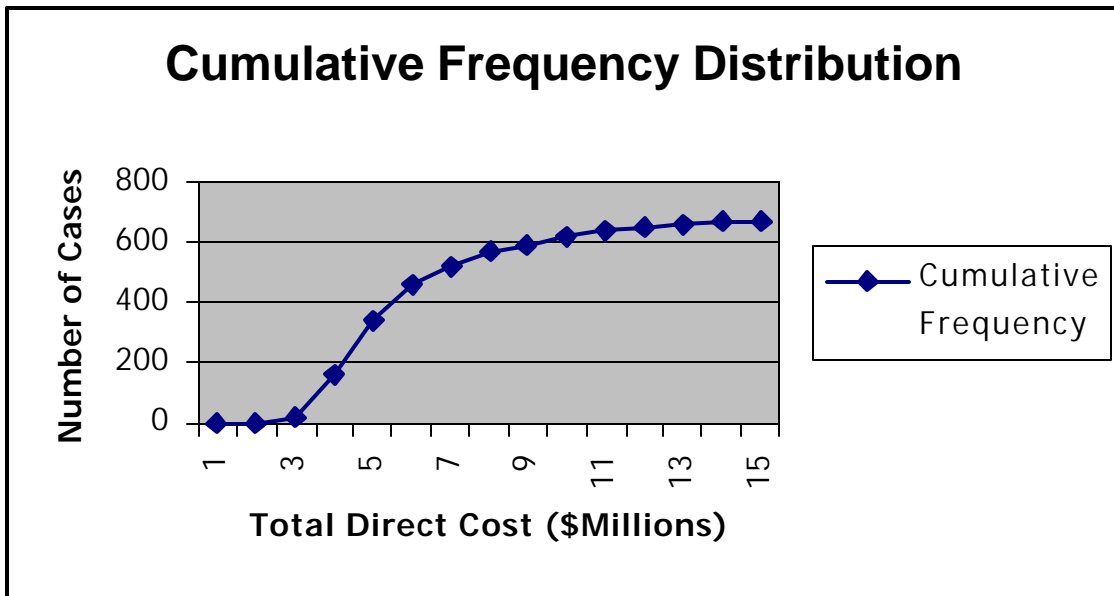


Figure 6: Cumulative Frequency Distribution of Total Project Costs

The total direct cost as shown in these plots represents the sum of the ANN model's prediction for the seven Unifomat categories exhibiting variation as a result of the

manipulation of input variables. Total direct cost for all Unifomat system categories can be obtained by adding \$2,225,831 to each value, to account for the missing eight Unifomat categories which remained constant over all samples (see Table 5).

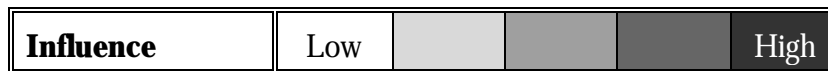
Extraction of Cost Drivers from the ANN Cost Model

The primary motivation for using ANNs to model the cost estimating process is their capacity to internally represent complex, non-linear relationships between input and output variables. Given a sufficiently large number of input variables, performing a traditional regression analysis becomes computationally prohibitive when nonlinear relationships are present. An advantage of using ANNs is their capacity to approximate non-linear relationships between inputs and output variables with relatively little computational effort. During the training process, an ANN model continually refines its internal architecture to produce more accurate outputs by comparing its initial output to the desired or target output provided by the trainer and adjusting its connection weights to increase the accuracy of its output. ANNs with at least one hidden layer approximate nonlinear regression (NeuralWare 1996), and therefore have the capability to model complex relationships among large numbers of variables, a feat which is typically computationally prohibitive using traditional regression techniques. After training, an ANN model can be manipulated to extract information about the relative importance of input variables in predicting output variables, which was the second objective of this research.

In order to utilize the ANN model as a source of information for cost drivers, the best-performing ANN model was exposed to an identity matrix as a means for extracting composite weights influenced by each input over the full network architecture. The identity matrix served as a “recall” set of cases, in which only one input neuron was activated at a time. The output for each identity case was an approximation of the relative importance of the input variable activated for that case. The set of outputs for the recall set was “unsquashed” to restore cost proportions, summed, and sorted in ascending order to provide a prioritized list of input variables in terms of their importance in driving final project cost. The resulting list is shown in Figure 7. Influence of each variable on system costs and total direct cost is shown using shading.

Figure 7: Prioritized List of Input Variables Representing Project Cost Drivers

	SUBSTR.	SUPERSTR.	EXT CLSRE	ROOFING	INT CONSTR	CONVEYING	HVAC.	INFLUENCE ON DIRECT COST
Number of Stairwells								
Floors Above Grade								
Exterior Doors								
Roof Area								
Number of Elevators								
Footprint								
Floor to Floor Height								
Heating Load								
Exterior Window Area								
Perimeter								
Cooling Load								
Exterior Wall Area								



According to the influences derived from the trained ANN model, the three most influential variables driving total direct cost were building perimeter, cooling load, and exterior wall area. In the PACES model, the cooling load and exterior wall area variables are also dependent on the number of floors above grade, building perimeter, and floor to floor height. While the ANN model did not learn the dependencies between these variables, it did predict the most influential variables in terms of total direct cost.

DISCUSSION AND CONCLUSIONS

The research results presented in the previous sections lead to two application-related questions. First, what are the potential impacts of providing knowledge to project stakeholders about cost drivers and cost-related risk? Second, how applicable is the methodology developed in this research to other types of facilities besides 1+1 dormitories, and what are the issues relating to validation of the methodology? The following sections address each of these questions in turn.

Potential Impacts of Cost Driver and Risk Information on the A/E/C Industry

The ability to extract information about the primary factors driving project cost holds promise for improved management of the planning and design processes. The knowledge of cost driver factors will enable project owners to make informed decisions when specifying initial design parameters, and to prevent design decisions from being made casually when those decisions have the potential to strongly impact final project costs. With an awareness of the potential impacts of early design information, information from future projects can be compiled to provide real case data to develop improved ANN models.

In addition, knowledge of the cost/probability function will enable project stakeholders to estimate the risk of cost variance from the initial conceptual estimate, facilitating the budgeting process and helping to encourage “tighter” design control when the risk of cost variance is unacceptably high. Together, these project cost control tools can help to improve project performance in terms of cost by providing quantitative data previously available to project stakeholders only through experiential heuristics.

Applicability and Validity of the Cost Driver Extraction Methodology

While longitudinal field validation of the results of this research was outside the scope of this research, initial heuristic validation of the research outcomes supports the methodology underlying the process of cost driver extraction. Although this proof of concept was limited in scope to one specific type of vertically-constructed facility within the entire range of construction projects, the methodology demonstrated in this work is anticipated to be

extensible to other types of construction projects within the sector of vertical construction, as well as to other project sectors.

FUTURE RESEARCH

Three areas of additional work can stem from this research, including continued refinement of ANN performance beyond that achieved in this proof of concept, generalization of the cost driver extraction methodology to other types of construction projects, and the development of a project-specific methodology for identifying and prioritizing cost drivers.

Continued Refinement of ANN Performance

The first area of future research is to continue to experiment with additional configurations of ANN parameters and architecture, along with additional research into developing representative sample sets using clean, simulated data. While the purpose of this research was to demonstrate the concept of cost driver prediction using ANNs, additional refinement and development of theory could lend substantial benefit to this work. Future research will also include testing and validation of ANN models and cost driver extraction techniques using real data.

Generalization of Approach to Other Project Types

More research is needed to test the ANN approach to cost driver prediction in other project types besides military administration buildings and dormitories. Although significant differences are not anticipated for similar commercial type buildings, examples from other fields of construction such as industrial plants may pose a challenge due to the uniqueness of each facility.

Development of Project-Specific Cost Driver Identification Tool

One promising application of the ANN concept to project risk management is the potential for prediction of specific cost drivers for projects at the pre-design phase of work. While many project managers rely on years of expert experience to know what to watch for, the

development of knowledge-based systems for these applications has been unexplored. Using ANNs to undertake identification of project-specific cost drivers is a promising application for ANNs, extending the capabilities of generalized feature recognition for generic construction projects exploited in this research.

ACKNOWLEDGEMENTS

This research was sponsored by the U.S. Air Force Office of Scientific Research under a Graduate Student Summer Research Grant. Grateful acknowledgement for technical and facilities support is given to Delta Technologies Group, Inc., owner of the commercial license for PACES software. Thanks also to the Georgia Tech Research Institute for their financial support of this research, and to NeuralWare, Inc., owner of the commercial license for the NeuralWorks Professional II software used in this research.

REFERENCES

- Burns, T. J. (1997). Deputy Director, Construction Cost Management, HQ AFCESA/DC, Tyndall AFB. Informal interview. July 16.
- Feldman, S.E., and Lebiere, C. (1988). "The Cascade-Correlation Learning Architecture," *Advances in Neural Information Processing Systems*, v. 2, Morgan Kaufmann, ed.
- Gregory, R.A. (1997). Assistant Professor of Civil & Environmental Engineering, Georgia Institute of Technology. Informal interview. July 31
- NeuralWare. (1996). *Neural Computing* NeuralWare Technical Publications Group, Pittsburgh, PA.
- Rast, R.R. (1997). CEO, Delta Technologies Group, Inc. Informal interview, July 31.
- USC. (1995). 104th Congress, 1st Session, No. 2, Title 10, *USCode Armed Forces* as amended through 12/31/94. Committee on National Security, HoR, March 1995. Page 1047, Section 2853, Authorized cost variations. Ch 169 – Military Construction and Military Family Housing.

Wasserman, P.D. (1989). *Neural Computing: Theory and Practice*. Van Nostrand Reinhold, New York.

Williams, L.E. (1997). "Data Simulation Supporting Range Estimating for Researching and Developing Alternatives". USAFOSR Graduate Student Summer Research Report.

APPENDIX A:

EXPERIMENTATION WITH THE PACES MODEL TO GENERATE A REPRESENTATIVE SAMPLE SET

Given the ability of ANN models to generalize, i.e., provide reasonable and accurate outputs to inputs not previously seen, we sought to develop a training set of examples which was minimal in size but still representative of the whole population of cases. Since generating cases using the PACES model was labor-intensive, determining which cases were representative was important to effectively utilize the ANN's generalization capabilities and to avoid the need to generate more cases than necessary to train the ANN.

As discussed in *Generation of a Representative Sample Set* we manipulated three independent variables to generate a representative population of data: number of floors above grade, building perimeter, and floor-to-floor height. Initial experimentation with PACES also involved varying the number of floors below grade, from a minimum of zero (no basement) to the model maximum of two. PACES experimentation was carried out iteratively, with parameters varied one at a time and the results plotted to determine the effects on total direct cost.

The first variable manipulated was number of floors, both above grade and below grade. Example cases were simulated in PACES for various combinations of floors above and below grade, as shown in Table A.1. The resulting impacts on total cost for all cases are shown in Figure A.1, and Figure A.2 shows the cases with no basement plotted by Unifomat system category. By plotting total direct costs in terms of component Unifomat system costs, the effects of varying the independent parameter could be seen, and explanations could be generated.

Table A.1: Simulated Combinations of Floors Above and Below Grade

Above	Grade:	1	2	3	4	5	6	7	8	9	10
Below	0	X	X	X	X	X	X	X	X	X	X
	1	X	X	X	X		X		X		X
	2	X	X	X	X		X		X		X

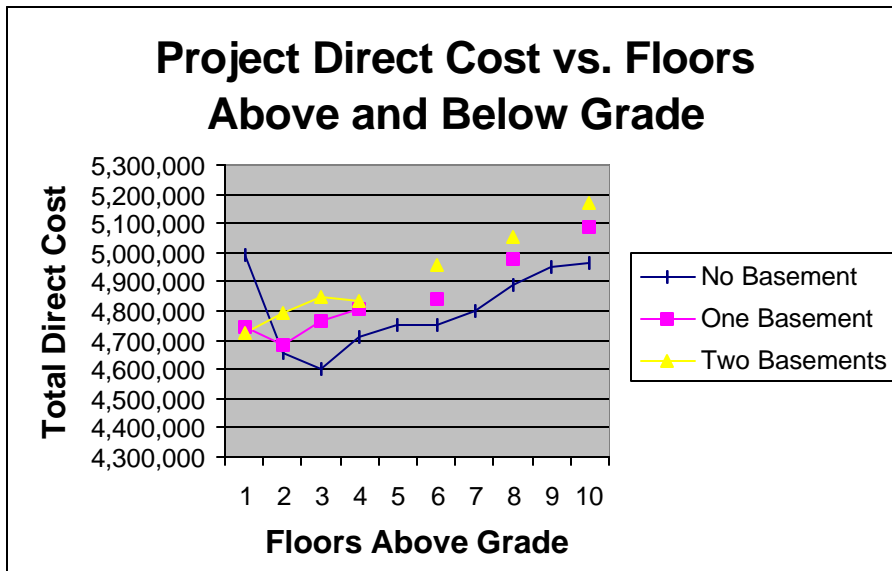


Figure A.1: Impacts on Direct Cost of Varying Number of Stories

For configurations with one and two basements, cases for 5, 7, and 9 stories above grade were omitted due to the relatively linear nature of the curve in this region. These omissions are an example of how the number of samples can be reduced based on knowledge of the population behavior. For the number of stories parameter, the behavior of the curve was based largely on the number and type of elevators in the region above three stories, as shown by the Conveying Systems cost in Figure A.2. Below three stories, total building cost is governed by a reduction in substructure and roof system costs due to the reduced footprint of the building. Above five stories, the reduced footprint of the building forces the number of elevators down to one, cutting the cost of conveying systems respectively. Above six stories, however, the type of elevator required shifts from a hydraulic system to a geared traction system, as evidenced in the building assemblies portion of the PACES model. This switch in system also increases conveying costs for buildings with six or more stories.

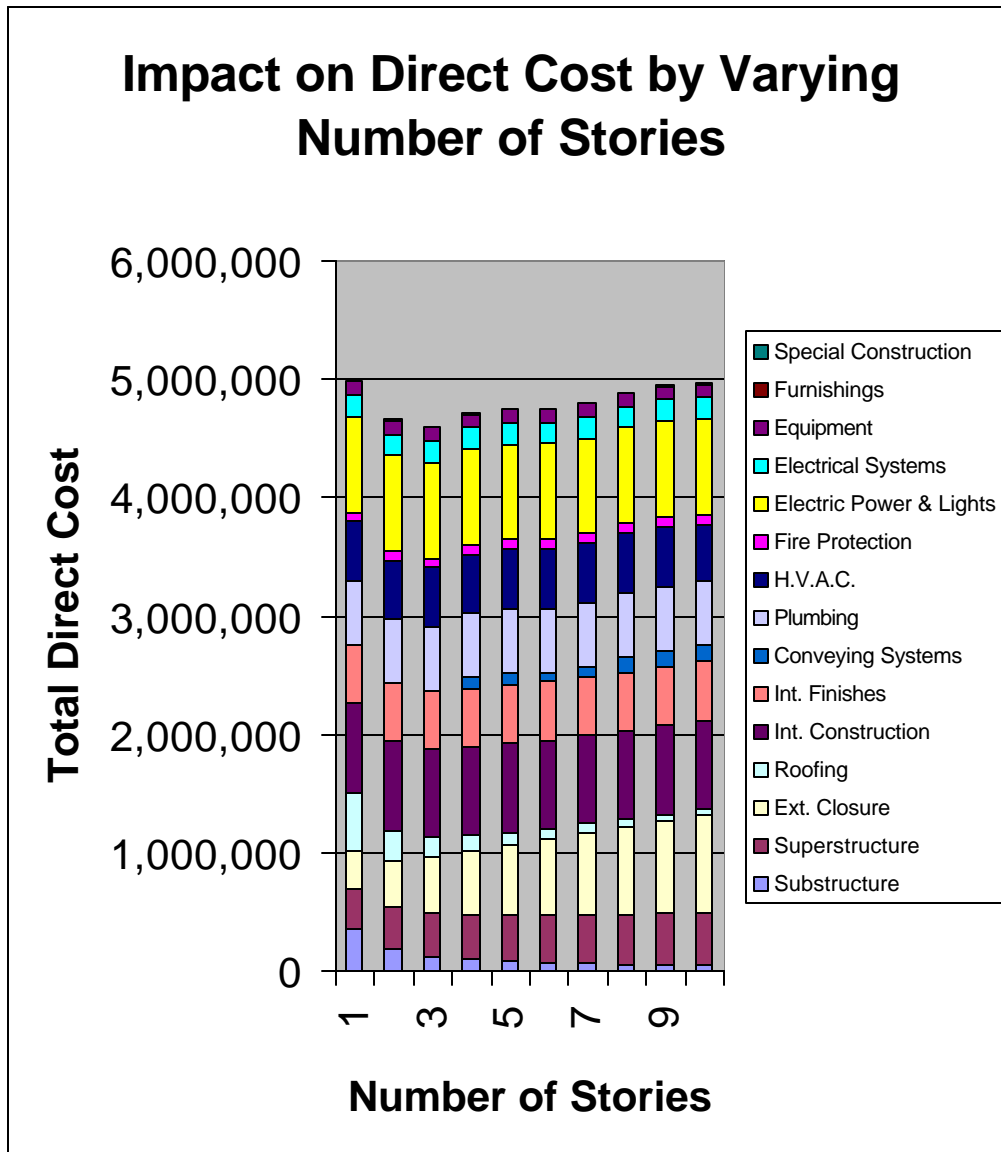


Figure A.2: Direct Cost Plotted by Uniformat Category

The next variable to be manipulated was the floor-to-floor height for facilities with no basements. We anticipated that the variance in total cost would be governed by exterior closure cost, and that the shape of the curve would reflect a linear increase in cost. We sampled floor-to-floor heights between 8 and 20 feet, in increments of four feet. Results of the sampling are shown in Figure A.3. As anticipated, the variance was linear.

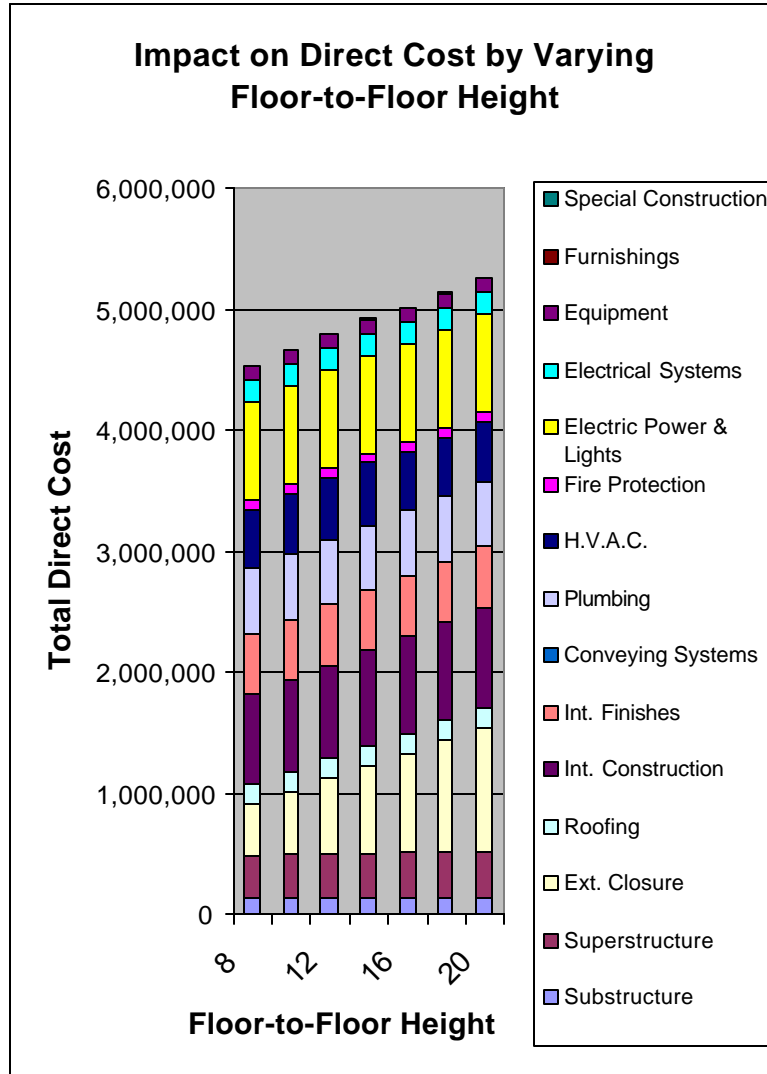


Figure A.3: Direct Cost as Influenced by Floor-to-Floor Height

Further investigation showed that the linear variance in direct cost due to floor-to-floor height plus the non-linear variance caused by changing the number of stories resulted in a cost curve similar in shape to the number of stories cost curve, only slightly more severe (Figure A.4).

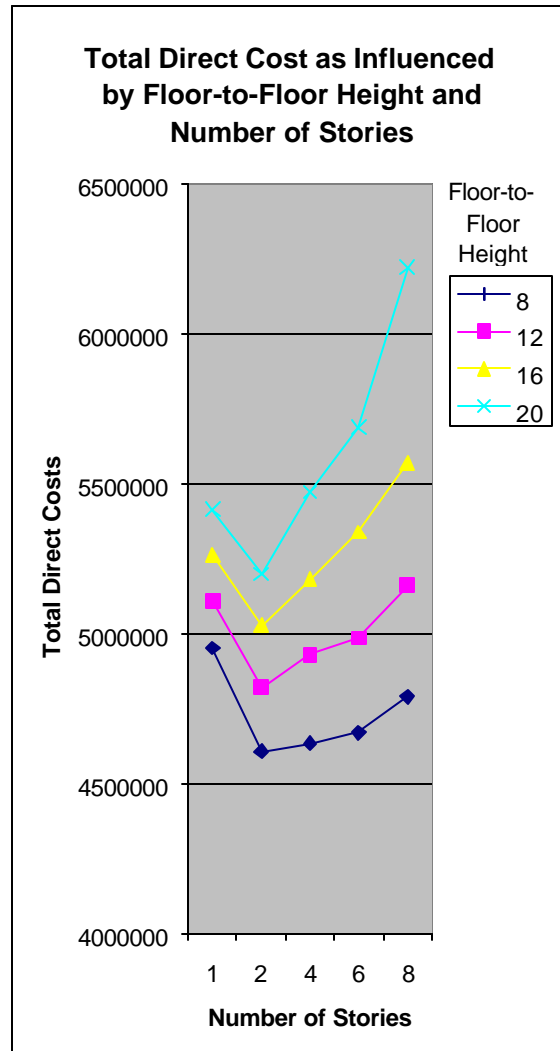


Figure A.4: Cost Variations with Change in Floor-to-Floor Height + Number of Stories

The third independent variable to be manipulated was building perimeter. While building perimeter is not strictly an independent variable in the PACES model, it can be independently manipulated as desired, or the default value calculated by the model can be accepted forthwith. To assess the impact of building perimeter on total direct costs, both alternatives were investigated. Simulations were run for numbers of floors above grade between 1 and 10, using the default building perimeter values generated by the model. For comparison, building perimeters of 2,000, 3,000, and 4,000 linear feet were run, using

numbers of stories ranging from one to three. The relationship derived between perimeter and cost is shown in Figure A.5.

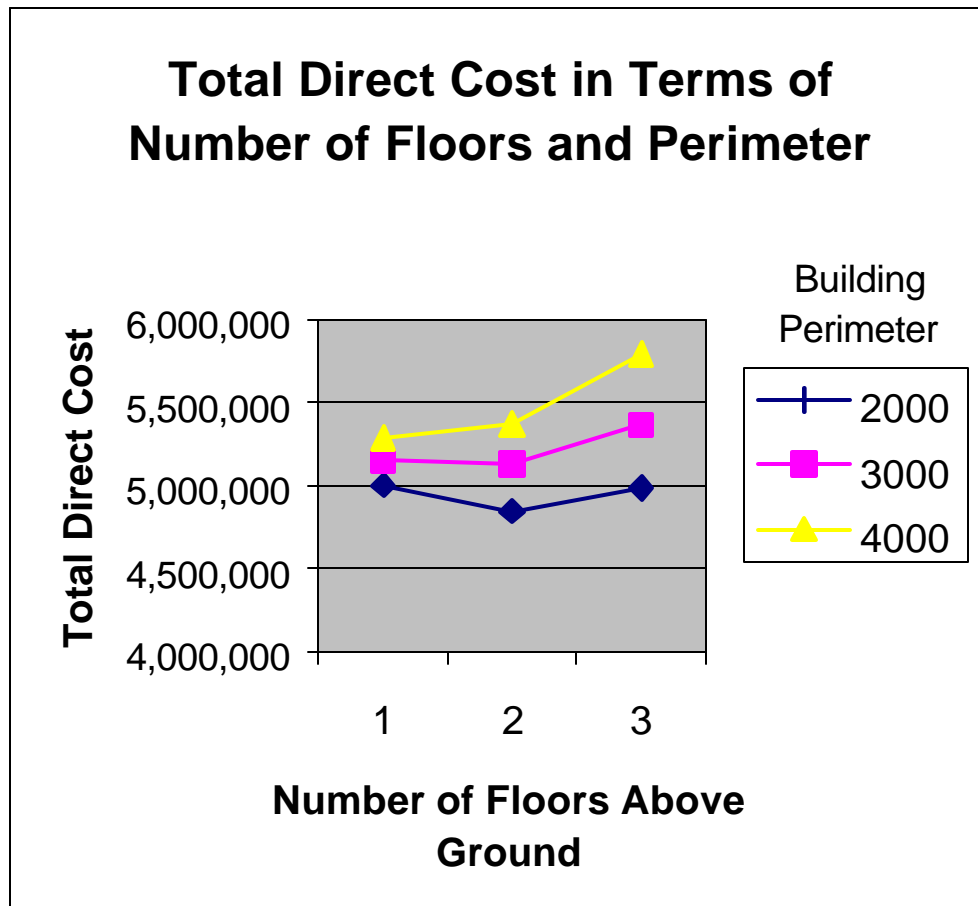


Figure A.5: Total Direct Cost as Influenced by Number of Floors + Building Perimeter

Further investigation into the relationship between building perimeter and number of stories was undertaken by breaking down total direct cost into Unifomat system categories for each of the nine permutations generated using the PACES model. Figure A.6 shows the outcome of the Unifomat breakdown, demonstrating that increases in cost are linear as a result of increasing building perimeter, primarily as a function of increasing exterior wall area/exterior closure requirements.

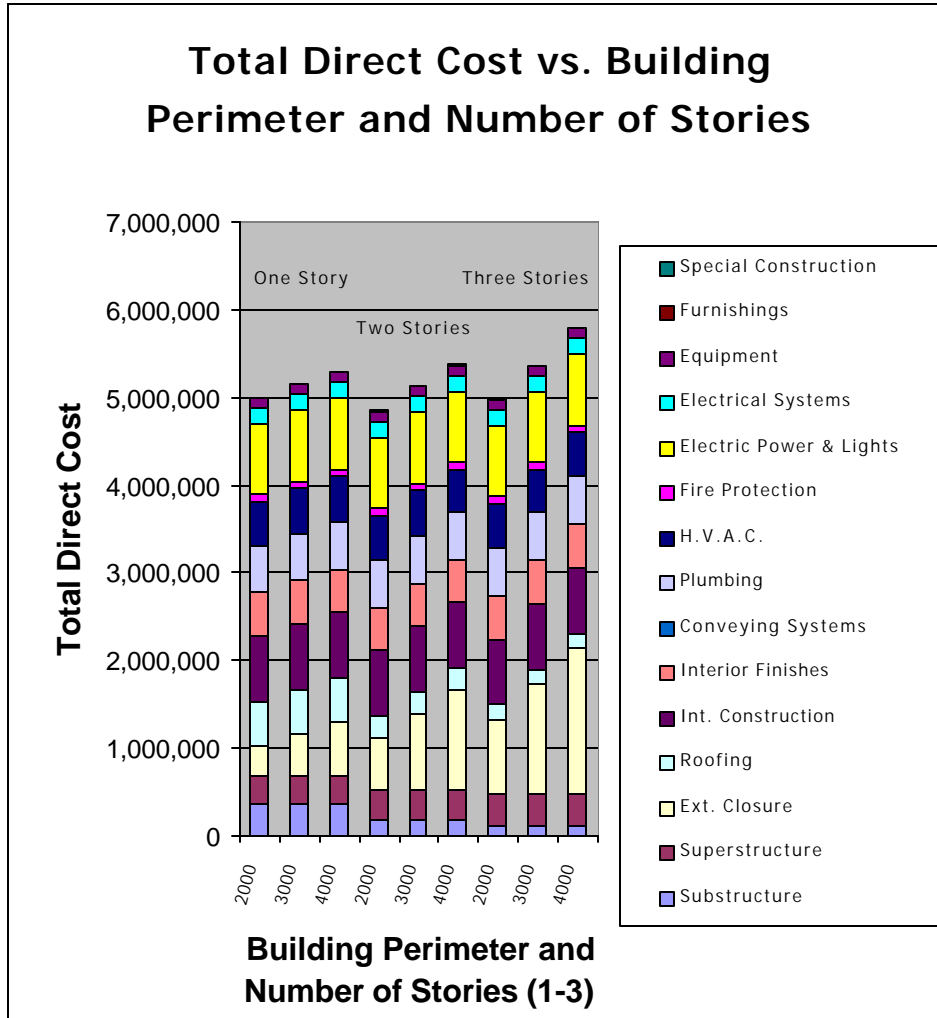


Figure A.6: Uniformat Breakdown of Cost Variations due to Perimeter Increases

By experimenting with the PACES model and plotting the results of varying the three independent variables, several conclusions could be drawn about the influence each of the variables has on total direct cost. First, the number of stories has a non-linear influence on total direct cost, due to the fact that at three stories, the complications of adding conveyance systems, combined with increasing exterior closure requirements, begins to offset savings in foundation and roofing costs realized by making the building footprint smaller. Second, increasing the floor-to-floor height of the building tends to linearly increase total direct costs due to an increase in exterior closure requirements. In reality, increasing the floor-to-floor

height should also have an impact on other costs such as structural costs, interior finishes, and conveying systems, but these factors are not programmed into the PACES model.

The third conclusion resulting from experimentation with the PACES simulated cases is that increasing building perimeter results in a linear increase in cost, again due to increasing exterior closure requirements. Since exterior closure comprises the largest portion of project costs in many cases, factors that influence the size and nature of this variable are likely to have a strong influence on overall project cost. This conclusion serves as informal validation of the ANN model results discussed in ***Extraction of Cost Drivers from the ANN Model***

APPENDIX B:

ARTIFICIAL NEURAL NETWORK BACKGROUND AND THEORY

This appendix seeks to explain the basic theory underlying Artificial Neural Networks (ANNs), in order to provide elementary background for readers unfamiliar with ANNs, their parameters, and appropriate applications. Readers interested in learning more about ANNs are referred to excellent introductory texts by Wasserman¹, NeuralWare², and Eberhart & Dobbins³.

PROPERTIES OF ANNS

Artificial neural networks are nonlinear computer models of the computational processes used by the human brain to make decisions. Most often used for tasks such as classification, signal processing, and nonlinear forecasting, ANNs are developed by “training” a matrix of weighted connections, each of which spans two of a set of processing elements (Figure B.1). Training has been achieved when the combination of processing elements and connecting weights responds accurately to different inputs by returning the associated desired outputs, each of which is called an “input-output pair” or I/O pair. A simple example of an I/O pair is a set of quantitative descriptors of a construction project and its final cost. The input vector might consist of factors such as {square footage, perimeter length, floor height, enclosure type, ..., n}, and the output vector could consist of one or more values, e.g., {final cost}, {substructure cost, structure cost, enclosure cost, ...}, etc.

¹ Wasserman, P.D. (1989). *Neural Computing: Theory and Practice*. Van Nostrand Reinhold, New York.

² NeuralWare, Inc. (1996). *Neural Computing*. NeuralWare Technical Publications Group, Pittsburgh, PA.

³ Eberhart, R.C. and Dobbins, R.W., eds. (1990). *Neural Network PC Tools: A Practical Guide*. Academic Press, San Diego, CA.

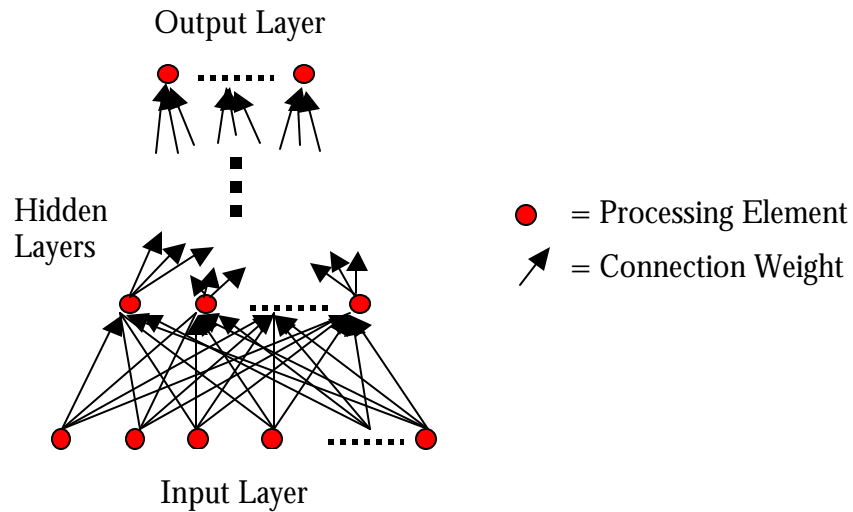


Figure B.1: The Architecture of an Artificial Neural Network

Developing or “training” the ANN requires repeated exposure to a set of such examples for the population of possible buildings. During each exposure, the ANN is shown the input, it generates its own output and compares that output to the desired output from the I/O pair, and self-adjusts its internal connection weights to produce an output more similar to the desired output. Given an adequate training set and sufficiently complex internal “architecture”, the ANN “learns” by self-adjustment over multiple iterations of the training set to generate the correct output for a given input, and even has the robustness to generate good approximations for inputs it has never seen before. This ability to generalize from known cases makes ANNs useful in situations where input data is likely to be confounded by noise or may be incomplete in some cases.

BACK-PROPAGATION ANNS

The specific type of ANN model used in this research was the Back-Propagation ANN. Back-propagation ANNs utilize a learning algorithm which changes connecting weights between processing elements based on the share of error each processing element or neuron contributes to the error at the next-higher layer. During each training iteration, an input vector is applied to the input layer of the ANN, where each input unit passes its activation

level through all connecting weights to the next layer. At the next layer, each processing element sums the product of the weights from all connected elements in the previous layer times the activation of the element corresponding to each weight (Figure B.2). This feed-forward process continues until the output layer is reached, whereupon the activation of each output neuron comprises one element in the output vector.

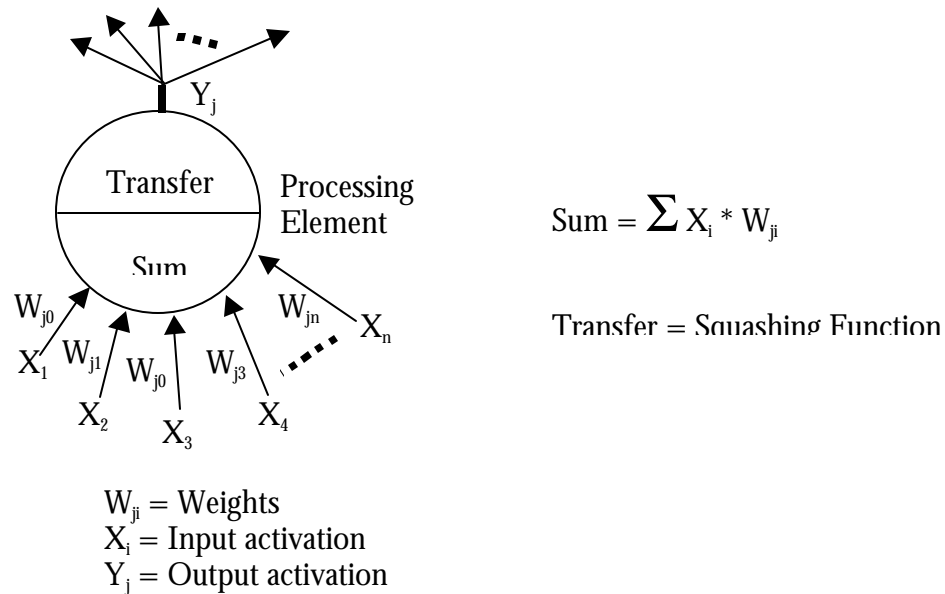


Figure B.2: Calculating the Activation of a Processing Element

At this point, the ANN's output vector (i.e., the activations of all output processing elements) is compared to the desired output vector provided as part of the I/O pair in the training set. The difference at each processing element is the total error for that neuron, and is used to adjust all weights connecting to that neuron. To adjust the weights of hidden layers for which there is no single output to determine error, the weights are adjusted as a product of the error for the next-higher layer times the changes in weights connecting that layer to the neuron in question. The reader is referred to any of the aforementioned references for a more detailed description of the back-propagation algorithm.

PARAMETERS OF THE BACK-PROPAGATION ANN

The back-propagation algorithm requires specification of several parameters. In this research, we elected to choose common values for each of the back-propagation parameters, with further experimentation reserved for future research.

Squashing or Transfer Function

First, the type of squashing function used is important to influence the way training occurs. The squashing function (see Figure B.2) is used to ensure that the sum of the weights times the activations of the connected neurons lies between 0 and 1. Depending on the type of ANN model selected, inputs to processing elements must typically be either between 0 and 1, or -1 and 1. Additional information about the role of squashing in processing input data is discussed in the section ***Developing an ANN Model for Cost Prediction***. For the purposes of this research, we selected a valid data range of 0 to 1, and the sigmoid squashing function to govern learning and calculation of neuron activations. The sigmoid squashing function is a commonly used squashing function in back-propagation applications.

Learning Rule

The second primary parameter of back-propagation ANNs is the type of learning rule used. Learning rules specify the algorithm used to modify weights during learning, and are discussed briefly in the previous section. Initial experimentation with different learning rules in this research showed that the Delta learning rule provided the most accurate models. Thus, the Delta rule was selected as the learning rule to be used in this proof of concept. The reader is referred to the aforementioned book by NeuralWare, Inc. for a detailed discussion of the algorithms used in the Delta rule as well as other learning rules.

Learning Coefficient

The third parameter required to train a back-propagation ANN is a learning coefficient. The learning coefficient governs how quickly weights can be changed over time, reducing the possibility of weight oscillation during training. Learning coefficients can be valued between 0 and 1, with lower values corresponding to slower weight changes and higher values

encouraging more rapid changes in weights. While high learning coefficients can encourage more rapid learning, they also increase the possibility of becoming “trapped in a local minimum” (Eberhart & Dobbins 1990), where the error minimization function used to train the network directs the network weights into a local optimum instead of the desired global optimum. A commonly used learning coefficient is 0.3, and this is the value used in this research.

Momentum

Momentum is the fourth parameter used for back-propagation learning. Like the learning coefficient, it influences how network weights are changed in an effort to reduce the possibility of being caught in a local error minimum. Momentum is applied to the weight-change algorithm by multiplying it times the previous value of the weight and adding the result to the new weight. Momentum can be valued between 0 and 1, and serves to encourage the weight change to move in the correct direction, as well as to suppress oscillations. A value of 0.4 is a common default value for momentum, and this value was used throughout the research.

Network Architecture

The final parameter to be varied in designing an ANN model is the architecture of the network. As shown previously in Figure B.1, ANNs always have an input and an output layer, and may have any number of hidden layers, each with many processing elements. The configuration of layers and processing elements is known as the architecture of the ANN, and is the primary determinant of network performance, given reasonable values for other network parameters. In practice, the number of hidden layers rarely exceeds two; some research indicates that the maximum number of hidden layers required to solve arbitrarily complex pattern classification problems is three (NeuralWare 1996). Since each neuron in a given layer is connected to all other neurons in all other layers, increasing the number of neurons and hidden layers increases the number of connection weights exponentially. Thus, from a computational perspective, training an ANN with a large number of hidden layers and neurons is much more computationally intense than training one with fewer layers or neurons.

On the other hand, neural networks with too few processing elements have insufficient complexity to adequately represent the relationships between inputs and outputs. One approach to determining an optimal number of processing elements is the cascade correlation algorithm (Fahlman & Lebiere 1990). In this algorithm, an ANN model begins with a single hidden processing element and adds new neurons while iterating through the training set until performance is no longer improved by adding new hidden units. At this point, the architecture of the network can be considered to be optimal. Experimentation in this research using both the cascade correlation algorithm and trial and error for determining an optimal architecture provided the same results (see ***Development of an ANN Model for Project Cost Prediction***).

APPROPRIATE APPLICATIONS FOR ANNS

Artificial neural networks are most effective for problems of sufficient complexity such that algorithmic approaches are unknown or are computationally inefficient. ANNs have been described as nonlinear regression models (NeuralWare 1996) and are particularly effective for tasks such as classification, where the number of possible outputs is fixed and the distinguishing features of each input are subtle, multitudinous, or complex. ANNs have been effectively applied to domains such as video-based vehicle detection (Bullock et al. 1992), where identification of vehicles is a 1-of-N classification task involving tracking the position of an optically detected object through a framework over time. Other 1-of-N classification applications such as character and voice recognition have also proven viable using ANNs, where every input may be slightly unique or incomplete, and algorithmic template matching has proven to be computationally overwhelming (NeuralWare 1996). ANNs have shown some promise in tasks of forecasting, where they have been used to predict stock market trends, weather conditions, and other typically “unpredictable” phenomenon (ibid.).

APPENDIX C:

PACES EQUATIONS FOR INPUT PARAMETER RELATIONSHIPS

The equations used to simulate the relationships between input variables in generating the comprehensive data set were obtained from the PACES Building Model Report, v. 3.10. Copies of the associated pages from the model report are attached, with relevant equations highlighted. In adapting these equations to the specific project scope (see Table 1), the equations describing the parameters listed in Table 4 were reduced to functions of the three independent variables manipulated to generate the comprehensive sample set.

Specific values obtained from PACES in order to make the calculations location-specific were as follows:

Atlanta Weather Region = 8

Atlanta Humidity Ratio $W_o = 140$

Percent of Building Cooled CPC = 70%

Calculations are included in the file “Comprehensive Data Set.xls” , included as part of this packet. Hard copies of the calculations and derivations are on file with the author of this report, and are available by request.