Improving Project Management Decision Making by Modeling Quality, Time, and Cost Continuously

Matthew J. Liberatore and Bruce Pollack-Johnson

Abstract—**The value of a project to a client can be measured in part by the level of quality associated with the completed project. Quality is acknowledged to be an important component of project management, but its joint relationship with time and cost previously has not been modeled. This paper introduces the notion of a quality function for individual tasks and uses the functional form of the bivariate normal (after also evaluating a bivariate logistic form), to model quality at the task level. Using real data from two case studies, a translation agency and a software development company, the quality function is specified and incorporated into a mathematical programming model that allows quality to be explicitly considered in project planning and scheduling. An alternative model formulation leads to the creation of quality level curves that enable managers to evaluate the nonlinear tradeoffs between quality, time, and cost for each of the example projects. The results of these analyses lead to specific decisions about the planned values for these three fundamental dimensions at the task level and provide insights for project planning and scheduling that can be gained through improved understanding of the choices and tradeoffs.**

Index Terms—**Mathematical programming, nonlinear optimization, project management, project planning, project scheduling, quality management.**

I. Introduction

PROJECT management addresses cost, schedule, and performance targets while providing an outcome that satisfies the client. A measure of the value of the project to the client is the level of quality associated with the completed project. It follows then that important managerial decisions relate to the level of quality achieved for each of the project's tasks, since *in toto* the quality of the tasks defines the quality of the project.

The emphasis in project planning and scheduling has been on managing the relationship between time and cost, with an implicit assumption of a fixed level of quality that is seldom explicitly examined. However, in many situations there are alternative approaches for completing each task, each having its own time, cost, and quality considerations. Differences in quality can arise due to bids offered by competing subcontractors to complete specific tasks. Even different bids by the same subcontractor could imply different quality levels. For example, subcontractors might have some flexibility with time and cost that would result in different quality levels for the same task. This can also be true for alternative work plans offered in-house. For example, in completing a foundation for a building there are choices related to the depth of the excavation and the compressive strength of the concrete used. Each of the possible alternatives will achieve different levels of time, cost, and quality associated with this task.

In the field of project management, "Quality management has equal priority with cost and schedule management" [1]. This statement makes inherent sense, since project management is concerned with not only managing cost and schedule, but also the actual work completed in order to achieve the project goal. The quality of the work completed then is an important project outcome, since it directly relates to the value of the project deliverables. The Project Management Body of Knowledge (PMBOK) [2] has adopted the ISO 9000, clause 3.11 definition of quality as "the degree to which a set of inherent characteristics fulfills requirements" [3]. Quality issues must be addressed in both the management of the project and the product of the project [2]. Specifically, the PMBOK suggests that quality must be addressed throughout the project life cycle, beginning in the project planning phase and continuing through quality assurance and quality control [2]. Unfortunately, no guidance is provided in terms of how quality can be measured in a project context.

The construction industry has been concerned about quality for a long period of time and has conducted research to address this issue. A study by the Quality Performance Measurements Task Force of the Construction Industry Institute resulted in the quality measurement matrix [4] and led to the development of an approach to measure quality performance of engineer-procure-construct (EPC) projects [5]. The measures were tied to four total quality management components: customer focus, leadership, delivery, and employee empowerment. Under delivery, for example, the subcategories include cost, time, safety/health/environment, and product deliverables. The focus of this approach is on identifying and then tracking critical quality measures for each project phase. A project quality performance model based on empirical study of project control variables was developed for Hong Kong construction projects [6]. These variables are grouped under the headings of client, project, project environment, project team leader, project management action, and project procedure. The literature on quality issues and problems in the construction industry is summarized in [7].

Manuscript received September 30, 2011; revised May 23, 2012; accepted September 6, 2012. Date of publication January 9, 2013; date of current version July 13, 2013. Review of this manuscript was arranged by Department Editor S. Talluri.

M. J. Liberatore is with the Department of Management and Operations, Villanova University, Villanova, PA 19087, USA (e-mail: matthew.liberatore@villanova.edu).

B. Pollack-Johnson is with the Department of Mathematics and Statistics, Villanova University, Villanova, PA 19087, USA (e-mail: bruce.pollack-johnson@villanova.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TEM.2012.2219586

Several authors have attempted to develop methods to measure project quality. Paquin *et al.* [8] assess project quality by decomposing client satisfaction into a hierarchical structure of quality dimensions that are measured and aggregated using a multicriterion approach such as the analytical hierarchy process (AHP) [9] or multiattribute utility theory [10]. A limitation of the multicriterion approach for quality measurement is the necessity of identifying and evaluating a possibly unique set of quality dimensions for each task.

Pollack-Johnson and Liberatore [11] illustrate how the quality of a task option can be determined using the AHP. They extend the discrete time–cost problem by developing a mathematical programming model that determines optimal discrete options defined in terms of time, cost, and quality combinations for specific tasks to maximize overall project quality, whether determined by AHP or by some other method, subject to time, and cost constraints.

Icmeli-Tukel and Rom [12] present two models that integrate quality considerations into resource constrained project scheduling. In their study, quality is measured by the amount of time and money spent on reworking activities that do not satisfy specifications. Several methods of integrating rework times and costs into the models are proposed.

An alternative approach for measuring quality at the task level is to make a direct subjective assessment of quality [13], [14]. The approach proposed in [13] extends the standard time–cost tradeoff analysis by assuming that quality depends only on time and is independent of cost for a given time, a major limitation of their approach. Project quality is measured as the arithmetic or geometric mean of the quality of the activities or as the minimum quality of the activities. Isocurves are used to show the relationships between two of the three factors, with the other held at constant levels. Babu and Suresh [13, p. 321] state: "This paper illustrates the concept in a general project setting using linearity assumptions. More complex relationships could be modeled in a similar fashion to capture reality more closely."

Khang and Myint [14] applied the method provided in [13] to an actual cement factory construction project. In both papers, the quality levels associated with all normal times was set at 1.0, while the lower quality level at the crash time for each activity was measured relative to this level. The linearity assumption between quality and time was found to be problematic. The most difficult and controversial task was to assess the quality reduction associated with crashing. These authors suggested the need for a more holistic measurement of performance quality and a more realistic model to describe the relationships between the quality of individual activities (and therefore of the whole project) and the budget and time allowed.

In response to the suggestion in [14] and the limitations of the approach in [13] noted earlier, in this research, we model quality at the task level as a continuous nonlinear function of both cost and time.

In Section II, we introduce the notion of a continuous quality function for individual tasks, specified in terms of time and cost. We provide real data from two case studies, document translation and software development, to demonstrate how the quality function can be specified. We evaluate two different functional forms, the bivariate normal and the bivariate logistic, and select the bivariate normal as most representative of the quality–time–cost relationship. In Section III, we then incorporate quality at the task level into an analytic model for project planning and scheduling. Using our project examples, we show how the problem of maximizing the quality of the weakest link subject to time and cost limits can be formulated and solved. An alternative formulation leads to the creation of isoquality curves for these examples that enables managers to evaluate the tradeoffs between time, cost, and quality across each project. Using either formulation, the results lead to specific decisions about the planned values for these three dimensions for each project task as shown in Section IV. We also offer some managerial insights for project planning and scheduling in Section V that are derived from the analysis through improved understanding of these choices and tradeoffs. Section VI summarizes our conclusions and directions for future research.

II. Quality Function

A. Development

Our overall goal is to find a rich method for modeling the relationship between quality, time, and cost at the individual task level within a project, as well as at the overall project level. Our approach begins by formulating a model of the quality of each task as a function of the time and cost allocated to it. We assume that there is an entity (an individual or group within an organization, a subcontractor, etc.) that could complete a given task with different allocations of time and cost. We further assume that there is a quality function that assigns to each combination of time and cost a corresponding quality value.

We limit time t and cost c to reasonable values for the task at hand and assume that, within that domain, the quality function for a task has two basic properties.

1) Holding time constant, quality q is an increasing function of cost. Thus, if time is fixed, we assume that allocating more money to the task will increase quality.
2) Holding cost constant, quality is an increasing function of time. Thus, if cost is fixed, we assume that allocating more time to the task will increase quality.

If we normalize quality to be on a 0–100 scale, based on the two nondecreasing assumptions earlier, we would expect the graph to show quality being lowest at the corner of the domain with the smallest values of time and cost and highest in the opposite corner (the highest values of time and cost). For a fixed quality, we would expect a standard time/cost tradeoff curve that is decreasing and convex. Thus, to maintain the same level of quality, to reduce the time, one has to pay increasingly more money per unit, such as in standard project activity crashing [16]. This suggests a basic hill shape rising out of a plain, although we would only be interested in a one-quarter wedge of the hill (as in Fig. 1).

A familiar mathematical functional form that has this shape is the bivariate normal distribution in probability. We investigate using this functional form for the overall quality function for each task. The probability function is normalized by the area under the surface being 1; our quality function is normalized so

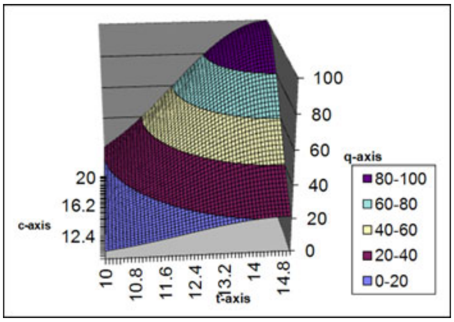

Fig. 1. Bivariate normal quality function.

that the maximum time μ_t and cost μ_c values considered reasonable correspond to a quality of 100, or some other constant indicating the maximum quality level possible K. The "standard deviation" parameters (σ_t and σ_c) give a measure of how slowly the quality drops from the top of the hill compared to the maximum values for time and cost, respectively. Thus, our resulting quality function is given by

$$Q(t,c) = Ke^{-\left[\left(\frac{t-\mu_t}{\sigma_t}\right)^2+\left(\frac{c-\mu_c}{\sigma_c}\right)^2\right]}. \tag{1}$$

The bivariate normal quality function is shown in Fig. 1. For generality and simplicity, we have normalized the initial constant to K and eliminated the $\frac{1}{2}$ in the exponent (which means that each σ in the quality function definition would be multiplied by $\sqrt{2}$ to be interpreted as the usual σ in the bivariate normal distribution). Note that if we hold either variable constant, the marginal graph for the other will be a bell curve (actually, a subset of a graph that is a constant multiple of a normal distribution curve).

For a given value of quality using the bivariate normal quality function, there is a nonlinear relationship between time and cost, generalizing the standard linear approach. This quality function is increasing with respect to both time and cost (holding the other constant), and should be able to represent the relationship between time, cost, and quality. Note that the version of the bivariate normal distribution upon which this functional form is based assumes independence of the two random variables. We have chosen this version for a simpler model, but the dependent version could also be used, with one more parameter, corresponding to the correlation/interaction between the variables.

In addition, we allow for the situation in which different entities could complete a given task. Here, each entity would have its own quality function. If those quality functions are graphed on the same time/cost/quality axes, then the overall quality function for the task that we want is the maximum, or the upper envelope, of the individual entity quality graphs. Clearly, if there are choices that provide different qualities for the same time and cost combination, the project manager will choose the highest quality option that is possible. This approach is consistent with the assumption of efficiency in new product development [15].

Another possible functional form meeting the two basic properties described earlier is a bivariate logistic function:

$$Q(t,c) = \frac{K}{1 + a_t e^{-b_t t} + a_c e^{-b_c c}}. \tag{2}$$

Its shape is somewhat similar to the bivariate normal but it tends to be more "squared off." We fit the bivariate logistic to the real data from the two case studies as described below, and found that it did not fit the data as well, and the results of the optimization problem did not fit the real situation as well, and so we used the bivariate normal for our quality function.

B. Estimation

The bivariate normal values μ_t and μ_c could be estimated as the maximum practical time and cost values for a given task. The values of σ_t and σ_c could be estimated by specifying the value of each variable (t_0 and c_0) that would achieve a specified quality value q_0 when the other variable is at its maximum. The σ values could then be calculated using the following formulas:

$$\sigma_t = \frac{t_0 - \mu_t}{\sqrt{-\ln(q_0/K)}}, \quad \sigma_c = \frac{c_0 - \mu_c}{\sqrt{-\ln(q_0/K)}}.$$

Alternatively, in situations where n bids, alternative work plans, or scenarios specifying levels of time, cost, and quality (t_j, c_j, q_j) have been received for a given activity, the four parameters of the bivariate normal function can be determined using nonlinear least squares estimation:

$$\text{Minimize} \sum_{j=1}^{n} \{Q(t_j, c_j | \mu_{t_j}, \mu_{c_j}, \sigma_{t_j}, \sigma_{c_j}) - q_j\}^2. \tag{3}$$

It is necessary to add the constraints:

$$\begin{aligned} \mu_{t_j} &\geq t_j, \quad j = 1, 2, \ldots, n \\ \mu_{c_j} &\geq c_j, \quad j = 1, 2, \ldots, n \end{aligned} \tag{4}$$

because the μ parameters are upper bounds on the values of the t and c variables.

The parameters of the bivariate logistic function can also be determined using nonlinear least squares regression, with no additional bounding constraints required.

C. Data Fitting

1) Translation Example: A Midwestern translation agency had received a large translation project that needed to be completed in at most 4 days. The project was too large for one translator to complete in this amount of time, so the project manager planned to divide the job between two translators. The project manager had two translators in mind, and, with our help, defined a quality scale that connects naturally to two different systems of measuring the quality of translation work. The Interagency Language Roundtable (ILR) provides one of these systems, and has four levels (4, 3, 2+, and 1+) [17]. Another system comes from the rubric used by graders of the American Translators Association (ATA) certification exams, which identifies "points" for weaknesses or errors in a translation, and categorizes a translation into four categories based on the number of points identified (from 0 to 46+) [18]. The hybrid

TABLE I
TIME, COST, AND QUALITY ESTIMATES FOR TRANSLATION PROJECT

Translator 1			Translator 2		
Deadline (days)	Cost ($)	Quality (0-44)	Deadline (days)	Cost ($)	Quality (0-44)
t	c	q	t	c	q
4	1435	41	4	653	32
3	1435	35	3.75	653	26
3	1750	41	3.75	800	32
3	2150	43	3.75	975	33
2.5	2150	37	3.5	975	26
2	2150	33			

TABLE II
RESULTS OF TRANSLATOR BIVARIATE NORMAL QUALITY FUNCTION ESTIMATION

	Translator 1	Translator 2
Constant [equation (3)]	50	44
SSE	1.89	1.26
Time mean (μ_t)	4.5086	4.3768
Time std. dev. (σ_t)	3.8223	1.2013
Cost mean (μ_c)	2150.0	975.00
Cost std. dev. (σ_c)	1655.1	671.10

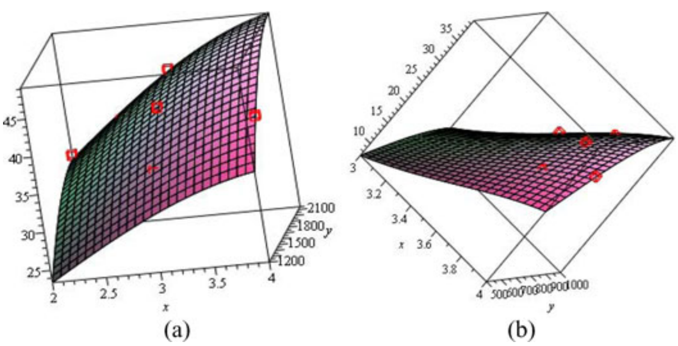

Fig. 2. Plot of translator 1 and 2's bivariate normal quality function with data. (a) Translator 1. (b) Translator 2.

quality scale system developed was on a 0–44 scale (44 being the highest possible quality) so that (roughly) quality levels that rounded off to 40 (to the nearest 10) would correspond to ILR Level 4, rounded to 30 would be Level 3, rounded to 20 would be Level 2+, and below 15 would be Level 1+. For the ATA rubric, the number of points can be subtracted from 44 to get a rough equivalent value on the 0–44 quality scale. After the subtraction 35–44 points corresponded to Strong in that system, 27–34 Acceptable, 14–26 Deficient, and 13 or less Minimal. Note that the categories correspond quite closely between the two systems, which helps to validate this approach.

The translation agency tended to establish a rate per word with each translator, but for rush jobs, the word rate could be increased, or a flat dollar bonus amount could be added, or a combination of these could be used. Thus the cost of each of the two tasks is divisible (continuous). Similarly, the deadline (time) is divisible. To reflect the reality of the translation world, time is measured in days, thinking of 1 day as 16 working hours (say, 8 am to midnight). If the project manager was negotiating on a Monday afternoon at 4 pm, then a deadline of three days would mean Thursday at 4 pm obviously, but 3.5 would correspond to Friday 8 am.

With these conventions and the specific translators in mind, the project manager estimated the quality of translation that could be expected for different combinations of deadline and cost for each of the two pieces of the translation (each piece being assigned to a specific translator). The results are shown in Table I. Note that Translator 1 has consistently higher quality than Translator 2; this will become significant later.

The four parameters of the bivariate normal quality functions for both translators were determined by applying (3) and (4) to the data given in Table I using Lingo's Global Solver [19]. For Translator 2 the constant K in (1) was replaced by 44 to fit the specific translation quality scale developed earlier. The results are shown in Table II. The SSE is very small (1.89) and all estimated quality values are within one point of the actual values, well within the measurement error of the scale itself. Fig. 2(b) provides a plot of Translator 2's quality function and actual values from Table II.

A similar procedure was used for Translator 1, but in using a constant of 44, the best SSE was 11.65. By using a constant value of 50, the SSE was reduced to 1.26, again with no deviations more than 1. The estimated parameters are given in Table II, with Fig. 2(a) providing a plot of Translator 1's quality function and actual values. In practice the quality function in for Translator 1 could be capped (truncated) at 44 since this is the maximum value on the translator quality scale, if necessary.

2) Software Development Example: As a second example of fitting a quality curve, we present a case study of an East Coast firm that develops software applications for clients. The process typically involves gathering the customer requirements (after making sure the rough, likely cost is in the feasible range for the client), then developing a quote for the project cost. If the quote is accepted, staff are allocated, there is an initial team meeting, the main database and software configuration and testing are pursued in parallel (as well some tasks that are outsourced), then these pieces are integrated and finalized, and there is a final task of analyst testing before the software is sent to the client for their testing.

In working with the manager responsible for these projects, we decided to base the quality level on the concept of expected rework cycles resulting from client testing. Rework based on the client changing their requirements is not considered a quality issue, but rework needed to meet those requirements is. From the point of view of the software development company, clearly rework cycles based on customer testing are damaging to their reputation and future thriving as a business. The manager indicated that their goal as a business is to achieve a high level of reliability, the major component of which is avoiding client rework cycles. To define a 0–100 scale, then, the manager estimated the expected damage from rework cycles, with a client rework cycle costing 20 points on the 0–100 scale for each substantial issue discovered. For example, if the time and cost allocated to a task were estimated to lead to an 80% chance of a client rework cycle for a single major issue, the estimated quality would be 100 – (0.8) $*$ 20 = 84.

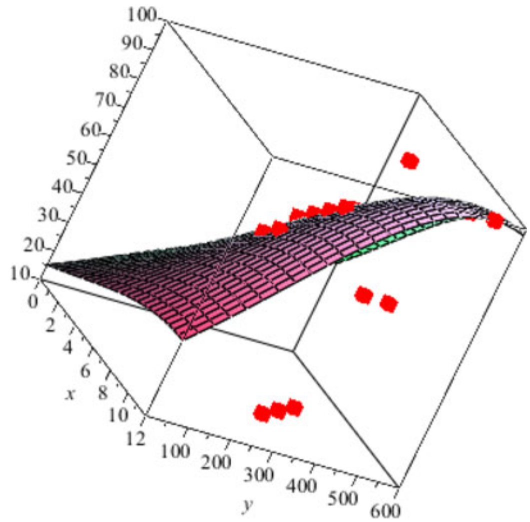

Fig. 3. Fit for task 4 (standard configuration and testing) showing upper envelope.

The process used to fit the quality curve was similar to the translation example above, but was more complicated because for some tasks there were more than one category of staff person who could perform the task, with different time, cost, and quality characteristics. This required the application of the upper envelope concept discussed earlier. If some time, cost, quality combinations were too inefficient, they were eliminated, and the curve was fit to the upper envelope points only. Fig. 3 illustrates this for one of the tasks. The data points under the curve were removed when fitting the model, since the employee they represent would not be chosen for that task (the quality is too much lower than the other ways of performing the task for the same time and effort). Note that such information has significant management implications, and suggests consideration of an intervention. The results of the fits for the seven tasks are shown in Table III.

Note that tasks 2 and 3 (and 0 and 8, the dummy start and finish activities) have fixed time and cost, so no decision needs to be made about them.

D. Comparison With the Babu and Suresh Approach

The approach of Babu and Suresh [13] and Khang and Myint [14] assumes quality is only a linear function of time. However, cost is also assumed to be a linear function of time, so that if we plot quality as a function of both time and cost, the 3-D graph would just be a line segment. Using the estimated data from Table I for Translator 2, the two endpoints of the line segment that is the implied quality function in [13], are a normal (time, cost, quality) of (4,653,32) and crash values of (3.5,975,26). Clearly, just assuming that Translator 2's quality can only lie on the line segment between the two points is much more restrictive than having the full continuum of possibilities of the curved surface, and does not reflect the full range of choices, but the fit between those two points is quite good.

Considering the software example, it is not clear how the method in [13] could be implemented, since the nature of the tradeoffs did not involve crashing. Instead, the empirical data indicated that later deadlines for the same number of billable hours yield higher quality. This relationship is captured by the bivariate normal quality function. However, the assumptions of [13] would have resulted in a line segment that had the same problem as in the translation example (not representing the full curved surface well), and the additional disadvantage of the line segment not fitting the surface well along its length (unlike the translation example).

III. Model Formulation

A. Maximizing Minimum Quality

Using the estimated quality functions for each project activity, we can determine the tradeoffs between quality, time, and cost. We start with standard assumptions for modeling projects: that the project network has no cycles, that the start activity (activity 0, a dummy activity) is the only activity that is not an immediate successor of any activity, and that the finish activity (activity $N+1$, also a dummy activity) is the only activity that has no successors. It is common to use predecessors rather than successors for project scheduling formulations of this type, but for this example, the formulation turns out to be much more concise and elegant using successors.

Define the following parameters and variables:

- t_i duration of activity i, for $i = 1,\ldots,N$;
- c_i cost of activity i, for $i = 1,\ldots,N$;
- q_i quality of activity i, for $i = 1,\ldots,N$;
- S_i set of activities that are immediate successors of activity i, for $i = 0,\ldots,N$;
- T_{UB} upper bound on the total project time;
- C_{UB} upper bound on the total project cost;
- s_i scheduled start time for activity i, for $i = 0,\ldots,N{+}1$;
- $t_{\min_i}$ lower bound on the duration of activity i, for $i = 1,\ldots,N$;
- $c_{\min_i}$ lower bound on the cost of activity i, for $i = 1,\ldots,N$.

Relevant quality measures from a project perspective could involve maximizing average quality, or maximizing minimum quality, of the tasks. We select the latter $Q_{\min}$ as our quality metric, since from a systems perspective if the project is viewed as an integrated set of activities, the quality of a project is only as high as its weakest link. $Q_{\min}$ is defined as follows:

$$Q_{\min} = \min_{1\le i\le N.} q_i \tag{5}$$

The quality–time–cost problem can be thought of as having three objectives, and can be modeled as a goal programming problem. In our formulation, we first maximize $Q_{\min}$ while setting an upper bound on total project cost and total project time. The nonlinear program is given as (6)–(18) below:

$$\text{Maximize } Q_{\min} \tag{6}$$

subject to:

$$Q_{\min} \le q_i,\ i = 1,2,\ldots,N \tag{7}$$

$$\begin{aligned} q_i = Q_i(t_i,c_i) = K * \exp\{-[(t_i-\mu_{t_i})/\sigma_{t_i})]^2 \\ -[(c_i-\mu_{c_i})/\sigma_{c_i})]^2\},\quad i=1,2,\ldots,N \end{aligned} \tag{8}$$

TABLE III
BIVARIATE NORMAL QUALITY FUNCTION PARAMETERS FOR SOFTWARE PROJECT TASKS

Task No.	Description	Pred.	Time Mean	Time Std. Dev.	Cost Mean	Cost Std. Dev.
0	Dummy	--	0	0	0	0
1	Requirements & Review	0	4.33	10.42	204.00	116.24
2	Setup	1	4.00	0	1045.00	0
3	Vendor Configuration & Testing	2	1.00	0	1000.00	0
4	Standard Configuration & Testing	2	12.00	12.82	525.00	520.65
5	Data Base Configuration & Testing	2	12.55	11.16	5493.10	4004.92
6	Development, Integration & Completion	3, 4, 5	1.54	1.88	56.16	52.06
7	Quality Assurance Testing	6	1.00	1.28	381.64	1246.51
8	Dummy	7	0	0	0	0

$$\sum_{i=1}^{N} c_i \leq C_{\mathrm{UB}} \tag{9}$$

$$s_0 = 0 \tag{10}$$

$$s_k \geq s_i + t_i \qquad \forall i = 0, \ldots, N \quad \forall k \in S_i \tag{11}$$

$$s_{N+1} \leq T_{\mathrm{UB}} \tag{12}$$

$$s_i \geq 0 \qquad \forall i = 1, \ldots, N+1 \tag{13}$$

$$t_i \geq t_{\min_i}, \quad i = 1, 2, \ldots, N \tag{14}$$

$$c_i \geq c_{\min_i}, \quad i = 1, 2, \ldots, N \tag{15}$$

$$t_i \leq \mu_{t_i}, \quad i = 1, 2, \ldots, N \tag{16}$$

$$c_i \leq \mu_{c_i}, \quad i = 1, 2, \ldots, N \tag{17}$$

$$q_i, t_i, c_i \geq 0, \qquad i = 1, 2, \ldots, N. \tag{18}$$

Equation (7) is a modeling device to obtain the minimum of q_i. The mathematical program is nonlinear due to (8), the bivariate normal quality function (the bivariate logistic function from (2) could also be used). Equations (9) and (12) set bounds on project cost and completion time, respectively. Equation (11) is needed to enforce the successor relationships among the tasks in the network. Lower bounds on the time and cost variable values are set by (14) and (15). These bounds are usually based on the smallest completion time and cost possible for each activity. Equations (16) and (17) reflect the upper bounds on the cost and time for each activity imposed by the definition of the quality function, although other values for the upper bounds could be considered. The independent decision variables are the time and cost for each project activity (which determine the quality for each activity). This problem can be solved using Lingo's global solver [19] and generalizes the standard time/cost tradeoff problem [16].

B. *Minimum Cost Formulation*

An alternative formulation of the problem considers minimizing total project costs with bounds on project completion time and quality. In the problem formulation (6)–(18), the objective in (6) becomes minimizing total cost or $\sum_{i=1}^{N} c_i$, thus eliminating (9), and then specifying a lower bound for project quality in (7), which becomes

$$Q_{\mathrm{LB}} \leq q_i, \; i = 1, 2, \ldots, N. \tag{19}$$

Analyzing the minimum cost model for different total project times T_{UB}, we can find the minimum cost possible that finishes the project within a given time and maintains a minimum quality of at least the lower bound. Note that this is closely related to the traditional crashing problem, because both work with a fixed level of quality. A set of isoquality (level) curves can then be constructed corresponding to different fixed quality levels, as will be demonstrated for our two examples below. These isoquality curves assist managers in evaluating the tradeoffs between time, cost, and quality across the project. This model then generalizes the standard time–cost problem by including quality considerations.

IV. RESULTS

A. *Translation Project Example*

1) Maximizing $Q_{\min}$: Returning to the translation example, consider the situation requiring a decision at 4 pm Monday concerning how much to offer to pay each translator, and what deadline to give each of them, to maximize the overall quality of the job (as defined by $Q_{\min}$, which the project manager agreed reflected reality well) and have the results in hand by noon Friday (equivalent to 3.75 days later, using our time scale), at a cost of no more than \$2400. This is a special case of the quality–time–cost model [(6)–(18)] with two parallel activities. Using the bivariate normal quality function estimates the model was solved using Lingo's Global Solver [19] with the results given in Table IV. We note in advance that all computational times for all runs using both the max $Q_{\min}$ and min cost models for both case studies was less than 0.05 s.

It was noted before that Translator 1 was of higher quality than Translator 2. Since the objective is to maximize $Q_{\min}$, the agency should put as much money and time into Translator 2 as possible to bring the quality up as high as possible. The lower bound on the cost for Translator 1 is \$1435, which means the upper bound for Translator 2 is \$2400−\$1435 = \$965, so both Translator 2 variables are at their upper bounds (\$965 and 3.75 days). The cost allotted to Translator 1 is the minimum

TABLE IV
SOLUTION OF TRANSLATOR PROJECT EXAMPLE USING BIVARIATE NORMAL QUALITY FUNCTION

	Translator 1	Translator 2	Project
Time	2.74	3.75	3.75
Cost	1435.00	965.00	2400.00
Quality	33.51	33.51	33.51

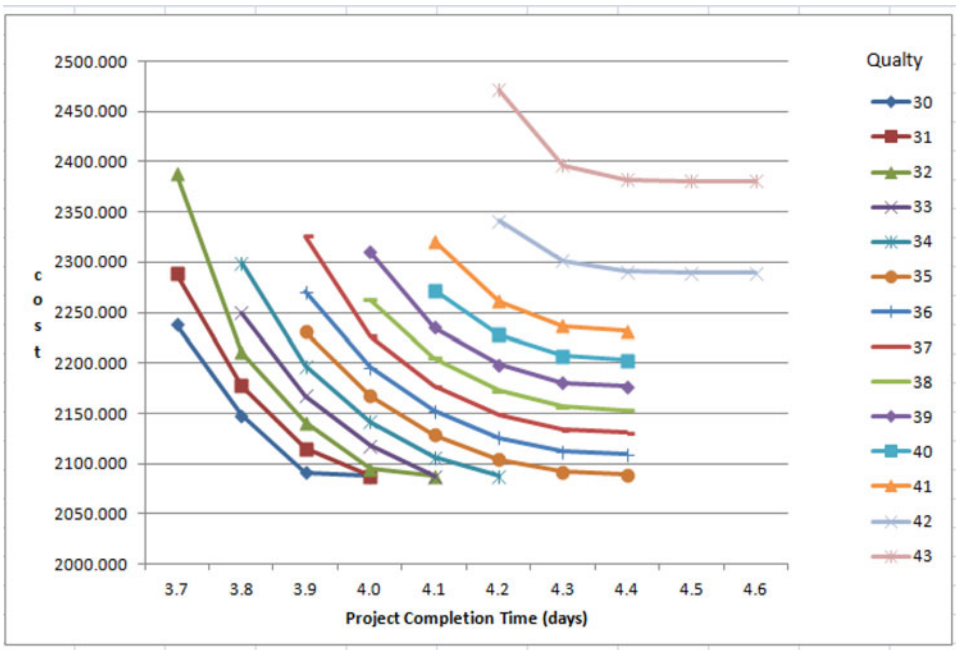

Fig. 4. Isoquality curves for translator example.

$1435, and the time is set at the value that will achieve the exact same quality as Translator 2 (33.51 on the 0–44 scale), which turns out to be 2.74 days (Wednesday noon). It is interesting to note that it is very possible that there may be no real advantage to the agency to set a tighter deadline for Translator 1 than the 3.75 days being given to Translator 2. In fact, the agency could make the deadline for Translator 1 to also be 3.75 days, without changing $Q_{\min}$ (this is an alternate optimal solution, clearly preferable to the agency!).

2) Isocurves: Using the minimum cost formulation discussed earlier, isocurves were constructed for the translator project and are shown as Fig. 4. Note that the isocurve for a higher quality level lies above and to the right of that for a lower quality level. Each isocurve in Fig. 4 provides the concave time–cost tradeoffs traditionally discussed in the project management literature. The isocurves in Fig. 4 provide a concise summary of the relationship among quality, time, and cost, and can be used by the project manager to make well-informed decisions about how to execute the project in terms of specific decisions about the work plans for all project tasks.

There are several interesting observations that can be gleaned from Fig. 4. First, assuming that the project must be completed by a deadline of 4 days, then a quality of 32 can be achieved at about the minimum cost of just under $2100. Starting from this solution, total cost will increase by the relatively small amount of around $50 for each unit increase in quality, and would seem to be worthwhile. However, the maximum achievable quality at the deadline is 39, regardless of cost. Therefore, to achieve higher levels of quality the deadline must be extended. Starting from a given quality and cost at the deadline, relaxing the deadline just a bit (0.1 day) can lead to an increase in quality by a point and no change or a reduction in cost as well. Second, to achieve the very high quality levels of 42 or 43 the project deadline must be extended and costs will increase significantly as well, indicating that these are not likely options. Third, the fastest time for project completion is 3.7 days but then quality can be at most 32 for a relatively high cost $2400. However, if time is increased by a small increment to 3.8 days, quality can increase to 34 with a reduction of cost to $2300, indicating extreme sensitivity to time on the low end.

B. Software Development Example

1) Minimizing Cost: Table V and Fig. 5 show the minimum cost problem solution for the software example, using $T_{\mathrm{UB}} = 19$, $Q_{\mathrm{LB}} = 69$, and the minimum values for the time and cost of each task set at 30% of the mean value (maximum). Recall that tasks 2 and 3 have fixed time and cost values, so have values that are whole numbers (and quality values that were set at 100).

TABLE V
SOFTWARE PROJECT MINIMUM COST SOLUTION FOR $T_{\mathrm{UB}} = 19$, $Q_{\mathrm{LB}} = 69$

Task No.	Description	Time	Cost	Quality
1	Requirements & Review	1.29897	141.780	69.0000
2	Setup	4.00000	1045.00	100.000
3	Vendor Configuration & Testing	1.00000	1000.0	100.000
4	Standard Configuration & Testing	12.0000	207.845	69.0000
5	Data Base Configuration & Testing	12.3180	3054.89	69.0000
6	Development, Integration & Completion	1.08295	27.1234	69.0000
7	Quality Assurance Testing	0.300060	114.493	70.7535

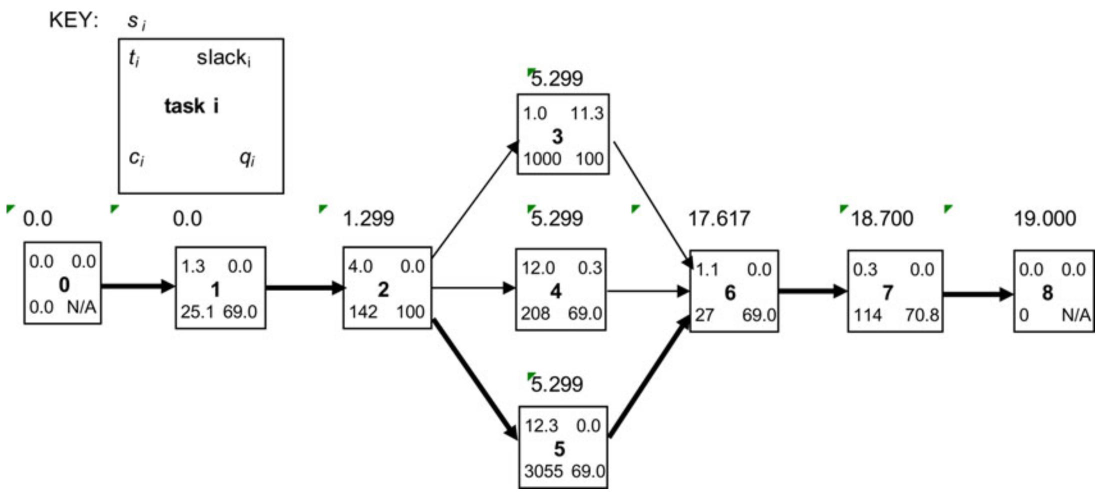

Fig. 5. Project network diagram of the minimum cost solution for the software example with $T = 19$ and $Q_{\mathrm{LB}} = 69$.

Note that most of the quality values are 69, which is the minimum quality level Q_{LB}. Tasks 2 and 3 have quality of 100, because their values were fixed. However, task 7 has a quality of 70.7535, which corresponds to the quality when the time and cost values are set at their minimum (note that the values are 30% of the mean values for both). This is a special case of a general result that can be easily shown:

Theorem: For the minimum total cost formulation described earlier, if the problem is feasible, then there will exist an optimal solution in which

$$q_i^* = \max\left\{Q_{\mathrm{LB}}, Q_i\left(t_{\min_i}, c_{\min_i}\right)\right\} \ \forall i.$$

a) Proof: See Appendix

Intuitively, this theorem simply says that the required quality level will be achieved exactly if possible, but if the minimum time and cost force a higher quality, the time and cost will be set at those minimum values (note that this applies to tasks 2 and 3 as well). If a task has slack at a given optimal solution that fits the theorem, its time value might be able to be increased, without increasing the cost, which would yield a higher quality for that task as an alternative optimal solution (and clearly preferable), similar to what we discussed for the maximizing $Q_{\min}$ model above.

The process of constructing a solution to the minimum cost model based on this theorem provides insights into the nature of the problem, and is briefly explained as follows. After making sure the problem is feasible, each task can first be examined individually to generate a candidate solution. If $t_{\min_i}$ and $c_{\min_i}$ yield a quality at or above the quality bound, then the time and cost are fixed at those values (as in the theorem). If not, the time is increased until the quality bound is attained if possible. If the time upper limit is reached first, then time is fixed there and the cost value is increased until the quality level is achieved. The resulting candidate solution meets the quality constraint, but may not fit within the time bound. In this case, a crashing-type problem is solved by decreasing the time of individual activities (and increasing the corresponding cost of each to maintain the same quality) until the time bound is achieved.

2) Isocurves: Fig. 6 shows the isoquality curves for the software development example. If, for example, the company wants at least a quality level of 90, they can inspect the curves for 90, 95, and 100, determine what seems optimal for each, and then apply judgment to decide whether or not the additional quality is worth the extra time and cost for the higher levels. In this case, at a quality level of 90, it seems the best time would be 19 weeks (as you reduce the time further, the cost increases much more significantly from that point), at a cost of a little under $7000. Khang and Myint [14] make a similar point in their analysis that for some quality levels, there seem to be budget threshold values of project times that are probably not worth the crashing effort. At 95, the best time seems to be 20 weeks, at a cost of a little under $7500. At 100, time would be 24 weeks, at a cost of about $8700. Depending on the circumstances, it does not seem that the added time and cost would be worth it to achieve the

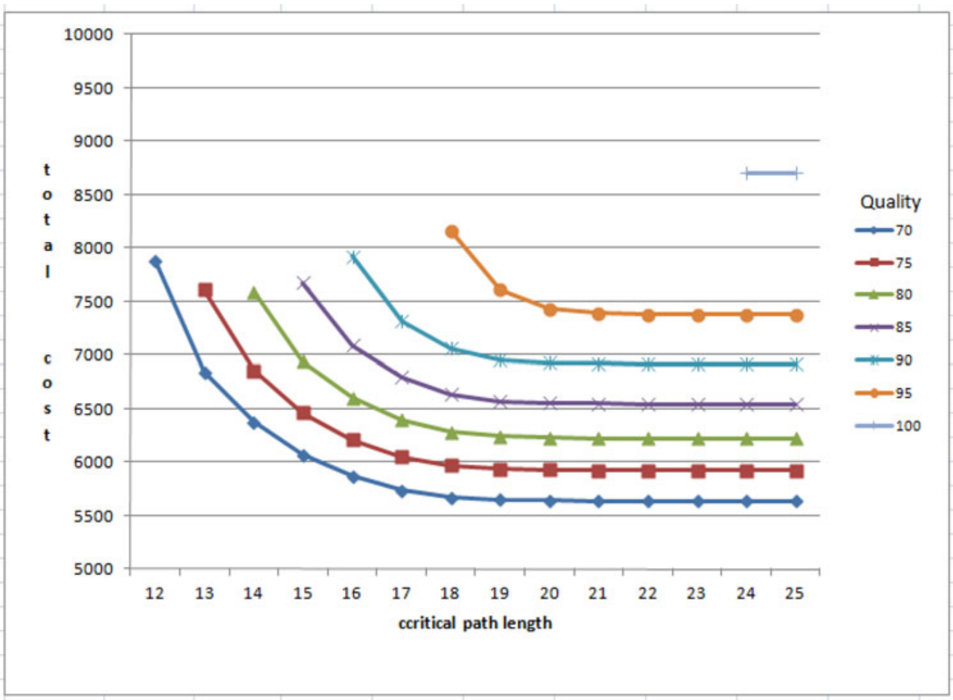

Fig. 6. Isoquality curves for software example.

quality level of 100. On the other hand, it might well make sense to increase the quality from 90 to 95, if it only means taking 1 week longer and costing $500 more. If desired, curves for 91 through 99 could also be generated for a finer analysis.

V. Managerial Implications

The methods presented in this paper demonstrate that practicing managers can include quality explicitly into their planning and scheduling decisions. The emphasis should be placed on estimating quality functions by obtaining multiple estimates of time, cost, and quality only for those tasks where there are clear choices to be made related to tradeoffs between these quantities that are likely to result in significantly different quality values and where the effort involved seems justified. In most cases, only four or five estimates are required for a given task, and fitting the quality functions and generating the isocurves are straightforward and can be done reasonably quickly. In our two case studies, the bivariate normal fit the data very well. However, in other situations, if the bivariate normal does not fit well, other functional forms satisfying the two properties we have specified should be considered. Furthermore, the collection of this data in itself could lead to insights about efficiency suggesting possible managerial interventions that might not otherwise be noticed.

We have provided applications of our approach using real data from case studies in two areas: translation and software development. Our approach is applicable in other areas as well. As discussed previously, the construction industry has demonstrated a strong interest in improving and managing quality. Our approach enables construction managers to addresses quality planning at the task level and to better understand how the decisions made affect overall project quality. Another important area of application is new product development, since managers must evaluate the tradeoffs inherent in allocating resources to develop higher quality products desired by the market with the pressure to enter markets on a timely basis to beat the competition. Moreover, in virtually any area of project management, explicit quality planning at the task level enables practicing managers to have a basis for tracking quality during project execution so that the planned quality levels can be achieved. Furthermore, our methods could be applied during the control phase of an ongoing project. For example, if there is a major cost overrun or time delay that makes achieving the original desired level of quality virtually impossible, the remaining tasks could have their time and cost allocations updated and be rescheduled using our techniques to minimize the reduction in project quality.

Without explicitly considering quality, inferior decisions may be made concerning the planned levels of effort for the various tasks. A key notion is that the quality of each project task is important and cannot be compromised without compromising the quality of the project itself. A conscious part of the project planning process should include identification of alternative work plans that consider the tradeoffs between quality, time, and cost. If total project time and budget are inflexible, the quality maximization model presented here can help project managers to select work plans for individual tasks that will increase overall project quality. If there is some flexibility in total project time and cost, the isocurve analysis can lead to important realizations such as that a significant increase in quality can occur with minimal impact on time and cost, or that time and/or cost could be appreciably improved with minimal effect on quality. The modeling framework proposed can provide insight on these issues and help to improve decision making in the project management planning and scheduling process.

VI. Conclusion and Future Research

The value of the project to the client can be measured in part by the level of quality associated with the completed project. Quality is acknowledged to be an important component of project management, but previously has received limited consideration in planning and scheduling. The implicit assumption behind standard time/cost tradeoffs is that some unspecified level of quality is maintained for each task. However, in many situations project managers must evaluate alternative options for accomplishing project activities, and these involve differing levels of time, cost, and quality. In such situations, it makes sense to analyze the relationship between cost, time, and quality, and decide on their levels for each project task that best achieves the project's objectives. We have introduced the concept of a quality function that represents the relationships between quality, time, and cost for each task. Using two case studies with real data, a translation agency and a software development company, the quality function is specified for each task and incorporated into a nonlinear programming model that allows quality to be explicitly considered in project planning and scheduling. An alternative formulation minimizes cost with bounds on project quality and completion time and leads to the creation of quality level curves. Both formulations can be very useful tools in making final project planning and scheduling decisions that explicitly model and incorporate quality.

An interesting insight is that by examining alternative optimal solutions additional quality for individual tasks can be obtained without additional cost. Once an optimal solution is obtained for either of our models, the solution can be checked to see if any tasks have slack and are not at their upper bound for time, since their times might be able to be increased (as long as the total project time never exceeds T_{UB}) without affecting the cost, and the quality of that task could be increased (as observed earlier in the translator $Q_{\min}$ example). If choices needed to be made between tasks whose time could be increased, those activities with the highest quality/time slope (slope of the curve sliced from the quality surface by the vertical plane corresponding to the current cost for that task) would be increased first, analogous to what is done with cost in the traditional crashing problem. One way to achieve such a superoptimized solution mathematically would be to extend our problem formulation by adding a second-level goal (where maximizing $Q_{\min}$ or minimizing the cost was the top-level preemptive priority) of maximizing the sum of the qualities of all of the individual activities. We leave these extensions for future research.

Objective functions other than maximizing the minimum quality over all tasks are possible, such as maximizing average project quality or a weighted average of minimum and average project quality, or goal programming models using the three objectives. A consideration of these varying model objectives and their impact on optimal solutions can form the basis of future research.

There are several limitations of our research. First, since the parameters of the quality function are expressed as crisp data, the effect of parameter uncertainty (if any) on the model's results is not determined. One approach is to use fuzzy or interval data to represent key parameters, and this would be an important extension to our work. Second, our model formulation considers direct costs that are associated with each task, but does not consider any project indirect costs. Indirect costs could address the overall administration of the project, including quality management. If the indirect cost is assumed to be a fixed charge, then its addition to the model will not affect the time, cost, and quality decisions for each task, but will increase the total project cost. In this situation, the indirect cost could be added to the total cost resulting from our model. However, if the indirect cost is assumed to depend on the total project duration, then it could affect the individual task decisions. One approach is to associate a base indirect cost with a target project duration, and then linearly scale the indirect cost based on the project's actual duration. We leave the development of this extension to future research.

Using the planned level of quality established for each task, future research can also investigate methods that can monitor and control quality during the course of the project, just as we now manage time and cost. Such methods will provide better balance in project management, emphasizing the nature, and quality of the work completed, not just its schedule and budget.

Appendix

Proof of Theorem

Suppose that $Q_i\left(t_{\min_i}, c_{\min_i}\right) > Q_{\mathrm{LB}}$ for some i, and suppose we have an optimal solution to the problem with $q_i^\circ > Q_i\left(t_{\min_i}, c_{\min_i}\right)$. Recall that we are assuming that the quality functions $Q_i(t_i, c_i)$ are all strictly increasing in t_i and c_i. If $c_i^\circ > c_{\min_i}$, then c_i can be lowered to $c_{\min_i}$ and t_i set to $t_{\min_i}$ (whether t_i was already there, or needs to be reduced to get there), resulting in a solution with a strictly lower cost that is still feasible. This contradicts the assumption of optimality, so $c_i^\circ > c_{\min_i}$ is not possible, implying that it must be the case that $c_i^\circ = c_{\min_i}$. Thus, $q_i^\circ > Q_i\left(t_{\min_i}, c_{\min_i}\right)$ can only happen if $t_i^\circ > t_{\min_i}$, but we can safely reduce t_i to $t_{\min_i}$ without becoming infeasible, since the precedence relationships and T_{UB} are maintained. This reduction would yield a solution with the same cost as the original optimal solution, and with $q_i^* = Q_i\left(t_{\min_i}, c_{\min_i}\right)$, thus satisfying the condition of the theorem. Thus, if an optimal (and feasible) solution had $q_i^\circ > Q_i\left(t_{\min_i}, c_{\min_i}\right) > Q_{\mathrm{LB}}$, then $c_i^\circ = c_{\min_i}$, and the q_i° solution with t_i° changed to $t_{\min_i}$ and $q_i^* = Q_i\left(t_{\min_i}, c_{\min_i}\right)$ will be an alternative optimal solution. This argument will hold for any i satisfying $Q_i\left(t_{\min_i}, c_{\min_i}\right) > Q_{\mathrm{LB}}$.

Now consider any i for which $Q_i\left(t_{\min_i}, c_{\min_i}\right) \le Q_{\mathrm{LB}}$ and suppose that we have a feasible solution to the problem with $q_i^\circ > Q_{\mathrm{LB}}$. If $c_i^\circ = c_{\min_i}$, then t_i can be feasibly lowered to some value, let's call it t_i^+ (at or above $t_{\min_i}$), at which point $q_i = Q_i\left(t_i^+, c_{\min_i}\right) = Q_{\mathrm{LB}}$, which must be an optimal solution for task i (since no other feasible values for c_i and t_i can have a lower cost), thus satisfying the condition of the theorem, and with the same cost as the original feasible solution. If $t_i^\circ = t_{\min_i}$ and $c_i^\circ > c_{\min_i}$, then we can similarly feasibly lower c_i to some c_i^+ (at or above $c_{\min_i}$), at which point $q_i = Q_i\left(t_{\min_i}, c_i^+\right) = Q_{\mathrm{LB}}$, which must be an optimal solution for task i (since no other feasible values for c_i and t_i can have a lower cost), satisfying the condition of the theorem, and with a strictly lower cost than the original feasible solution. If $t_i^\circ > t_{\min_i}$ and $c_i^\circ > c_{\min_i}$, then we can proceed in stages. If c_i^+ as defined earlier is greater than or equal to $c_{\min_i}$ then by lowering c_i to c_i^+ we have found a strictly better feasible solution satisfying the conditions of the theorem. If c_i^+ as defined earlier would be strictly *less* than $c_{\min_i}$, then we can lower c_i to $c_{\min_i}$ and lower t_i to t_i^+ as defined earlier, again resulting in a strictly better feasible solution satisfying the condition of the theorem. In all of these cases with $Q_i\left(t_{\min_i}, c_{\min_i}\right) \le Q_{\mathrm{LB}}$, we have shown that a solution with $q_i > Q_{\mathrm{LB}}$ and $c_i > c_{\min_i}$ cannot be optimal (since we could find a feasible solution with a strictly lower cost), and if an optimal solution with $q_i > Q_{\mathrm{LB}}$ and $c_i = c_{\min_i}$ exists, we have shown how to construct an alternative optimal solution that satisfies the condition of the theorem, thus completing the proof.

References

[1] H. Kerzner, *Project Management: A Systems Approach to Planning, Scheduling, and Controlling*, 8th ed. New York: Wiley, 2003.

[2] Project Management Institute, *A Guide to the Project Management Body of Knowledge,* 3rd ed., Newtown Square, PA, Project Management Institute, 2004.

[3] International Organization for Standards, ISO 9000:2000, 2000.

[4] J. Stevens, C. Glagota, and W. Ledbetter, “Quality-measurement matrix,” *J. Manage. Eng.*, vol. 10, no. 6, pp. 30–35, 1994.

[5] J. Stevens, “Blueprint for measuring project quality,” *J. Manage. Eng.*, vol. 12, no. 2, pp. 34–39, 1996.

[6] A. Chan, “A quest for better construction quality in Hong Kong,” *CIOB Construction Informat. Quarterly*, vol. 3, no. 2, pp. 1200–1213, 2001.

[7] A. Heravitorbati, V. Coffey, and B. Trigunarsyah, “Assessment of requirements for establishment of a framework to enhance implementation of quality practices in building projects,” *Int. J. Innov. Manage. Technol.*, vol. 2, no. 6, pp. 465–470, 2011.

[8] J. P Paquin, J. Couillard, and D. J. Ferrand, “Assessing and controlling the quality of a project end product: The earned quality method,” *IEEE Trans. Eng. Manage.*, vol. 47, no. 10, pp. 88–97, 2000.

[9] T. L. Saaty, *The Analytic Hierarchy Process,* Pittsburgh RWS Publications, 1996.

[10] R. Keeney and H. Raiffa, *Decision with Multiple Objectives: Preferences and Value Tradeoffs,* New York Wiley, 1976.

[11] B. Pollack-Johnson and M. Liberatore, “Incorporating quality considerations into project time/cost trade-off analysis and decision making,” *IEEE Trans. Eng. Manage.*, vol. 53, no. 4, pp. 534–542, 2006.

[12] O. Icmeli-Tukel and W. O. Rom, “Ensuring quality in resource constrained project scheduling,” *Eur. J. Operational Res.*, vol. 103, no. 3, pp. 483–496, 1997.

[13] A. J. G. Babu and N. Suresh, “Project management with time, cost, and quality considerations,” *Eur. J. Operational Res.*, vol. 88, no. 2, pp. 320–327, 1996.

[14] D. B. Khang and Y. M. Myint, “Time, cost and quality trade-off in project management: A case study,” *Int. J. Project Manage.*, vol. 17, no. 4, pp. 249–256, 1999.

[15] L. O. Morgan, R. M. Morgan, and W. Moore, “Quality and time-to-market tradeoffs when there are multiple product generations,” *Manuf. Serv. Oper. Manage.*, vol. 3, no. 2, pp. 89–104, 2001.

[16] P. Brucker, A. Drexl, R. Mohring, K. Neumann, and E. Pesch, “Resource-constrained project scheduling: Notation, classification, models, and methods,” *Eur. J. Operational Res.*, vol. 112, no. 1, pp. 3–41, 1999.

[17] Interagency Language Roundtable. (16 Sep. 2011). “ ILR skill level descriptions for translation performances,” [Online]. Available http://www.govtilr.org/Skills/AdoptedILRTranslationGuidelines.htm

[18] American Translators Association, ATA Certification Program Rubric for Grading, Version 2009. Available from the American Translators Association, Alexandria, VA.

[19] Lindo Systems Inc., Extended Lingo/Win 32, Version 13.0, Chicago, 2011.

Matthew J. Liberatore received the B.A. degree in mathematics and the M.S. and Ph.D. degrees in operations research, all from the University of Pennsylvania, Philadelphia.

He is currently the John F. Connelly Chair in Management and the Director of the Center for Business Analytics in the Villanova School of Business at Villanova University, Villanova, PA. He previously taught at Temple University and held positions at RCA and FMC Corporation. At Villanova University, he previously served as Chair of the Department of Management and as Associate Dean. Dr. Liberatore has published more than 80 peer-reviewed articles in operations research, project management, information systems, health care management, and research and engineering management. His current research focuses on modeling quality in project scheduling, analytics investment (with IBM), health care operations, and supply chain planning.

Dr. Liberatore currently serves on the editorial boards of *Computers & Operations Research,* IEEE TRANSACTIONS ON ENGINEERING MANAGEMENT, and the *American Journal of Mathematical and Management Sciences.* He is a member of INFORMS, the Decision Sciences Institute, and the Project Management Institute.

Bruce Pollack-Johnson received the B.A. degree in sociology with a minor in education from Brandeis University, Waltham, MA, the M.A. degree in applied mathematics from Temple University, Philadelphia, PA, and the M.S. and Ph.D. degrees in operations research from the University of Pennsylvania, Philadelphia.

He has taught at Oberlin College, and is currently an Associate Professor of mathematics and statistics at Villanova University, Villanova, PA. He has published dozens of papers in project management, forecasting, educational modeling, and on teaching applied mathematics, as well as three editions of a two-volume text on business calculus and finite mathematics (partially funded by grants from FIPSE, NSF, and Prentice Hall).

Dr. Pollack-Johnson’s current research is on modeling quality in project scheduling (for which he and Dr. Liberatore received the Best in Contribution to Theory award at the Northeast Decision Sciences Institute annual meeting in 2012) and analytics (with IBM). He is a member of INFORMS and MAA.