Journal of Cost Analysis and Parametrics, 7:129–147, 2014

ISSN: 1941-658X print / 2160-4746 online
DOI: 10.1080/1941658X.2014.929987

ICEAA
www.iceaaonline.com

Activity-Based Parsimonious Cost Systems

SHANNON CHARLES[1] and DON HANSEN[2]

[1]School of Accounting, University of Utah, Salt Lake City, Utah
[2]School of Accounting, Oklahoma State University, Stillwater, Oklahoma

Accurate product costing information is instrumental in effective decision making, especially for product related decisions, such as product mix and product emphasis. A body of literature suggests that activity-based costing plays an important role in providing accurate product costing information. However, activity-based costing systems are also more complex due to the number of cost drivers identified, compared to traditional single cost driver systems. We propose a modification to the activity-based costing system that reduces the complexity. Using the concepts of linearly independent and dependent vectors, it is shown that it is possible to identify parsimonious systems that simplify an activity-based costing system without loss of accuracy. The modification presented in this study will entice more firms to adopt an activity-based costing system and reap the benefits of increased product costing accuracy.

Introduction

An activity-based product costing (ABC) system is based on causal relationships among manufacturing activities and product costs. ABC systems produce cost assignments that are generally more accurate than unit-based systems based on plant-wide or departmental overhead allocation rates (Cooper & Kaplan, 1988). Evidence exists indicating this increased accuracy leads to better cost control, improved decisions, and increased firm profitability (Ittner, Lanen, & Larcker, 2002; Cagwin & Bouwman, 2002; Kennedy & Afleck-Graves, 2001). Yet, several researchers argue that ABC has not experienced as much diffusion as it merits given its claimed advantages (e.g., Anderson & Young, 1999; Innes, Mitchel, & Sinclair, 2000; Abernathy, Lillis, Brownell, & Carter, 2001; Cokins, 2002). The fact that many firms are not using ABC accentuates Gosselin's ABC paradox: "If ABC has demonstrated benefits, why are more firms not actually employing it?" (Gosselin, 1997). The nonusers of ABC, who have either considered ABC and rejected it or who are still considering it, identify *complexity* and *cost* as the major deterrents to its adoption (Innes et al., 2000; Krumwiede, 1998).

The complexity of an ABC system increases at an exponential rate as the number of activities required to produce a product increases. Larger, more complex ABC systems are more accurate than traditional product cost systems, but they are rarely sustained after a pilot project. Pryor (2004) offers three reasons for the lack of sustainability for larger, complex systems. First, there are so many activity or second-stage cost drivers, that the ongoing data required cannot be collected effectively. Second, the ABC bill of activities has so much data that nonfinancial users find it too complex to read, interpret, and use. Third, there are too many data locations for effective system integration.

Address correspondence to Shannon Charles, School of Accounting, 1655 East Campus Center Drive, Salt Lake City, UT 84112. E-mail: s.charles@utah.edu

Given the complexity of ABC is identified as one of the major deterrents to its adoption and/or sustainability; it seems reasonable that research efforts should focus on reducing complexity and, thus, work towards identifying simplified product costing systems. Product cost system simplification can be categorized into two major areas: *before-the-fact* simplification and *after-the-fact* simplification. *Before-the-fact* simplification addresses simplification at or before the implementation stage and affects *adoptability*. To reduce complexity before the fact requires the adopter to identify a full-blown ABC system and determine the level of accuracy of such a system *before* simplification can take place. This is difficult, at best, given the limited resources of most firms. Furthermore, there are no guidelines available for researchers and/or practitioners for undertaking such a task. Therefore, *before-the-fact* simplification is difficult to address and deserves the attention of future researchers. On the other hand, *after-the-fact* simplification is much more attainable and has been the main focus of simplification research thus far.

After-the-fact simplification models take an *existing* complex and well-specified system and use the information contained in this existing system to bring about the proposed simplification. The simplification thus occurs *after* a sophisticated activity accounting system is created. *After-the-fact* simplification is useful because it can ensure the *sustainability* of an ABC system. Although the simplified system may be less accurate than the parent system from which it is derived, "an approximately relevant ABC system is much more valuable than one that is precisely useless" (Pryor, 2004, 2). Because the accuracy of the more complex system is known, the accuracy of an approximately relevant ABC system can also be assessed.

The notion of an approximately relevant ABC system is based on the accuracy loss assumption. Prior research has stated that the accuracy of a system depends on minimizing aggregation, specification, and measurement errors, thus creating a complex, well-specified ABC system (Datar & Gupta, 1994). Aggregation error results from reducing the number of activity cost pools by adding costs of heterogeneous activities to form fewer and more aggregate cost pools. Specification error results from using the wrong cost driver to assign costs. Measurement error refers to incorrect assignment of costs to individual activities. Error is measured relative to an "unobservable" completely specified ABC system. Thus, it is generally assumed that moving from a complex, well-specified ABC system to a simplified, more aggregate system necessarily entails a loss of accuracy (Babad & Balachandran, 1993; Datar & Gupta, 1994; Homburg, 2001). Because of this assumption, simplification research has focused on minimizing accuracy loss as the number of drivers is reduced, while simultaneously considering information production costs (Babad & Balachandran, 1993; Homburg, 2001). No one yet has challenged the accuracy loss assumption.

The cost of an ABC system is the other major deterrent to its adoption and use. The cost of an ABC system can be divided into five identifiable costs:

1. activity identification costs,
2. first stage allocation costs,
3. activity driver identification costs,
4. ongoing data collection costs, and
5. complexity costs.

The first three costs are implementation and updating costs and are most effectively addressed by *before-the-fact* simplification. The last two costs are costs related to sustainability and are affected by *after-the-fact* simplification. Ongoing data collection costs are the costs incurred to measure expected and actual consumption of activities so that applied overhead costs can be computed (predetermined activity rates $\times$ the actual amount of activity consumed). Thus, activity costs and expected consumption patterns for each activity must be estimated and data must be collected for each driver so that overhead costs can be assigned to products as production unfolds. The more drivers there are, the more data collection needed. Complexity costs are the costs of foregone benefits because managers reject the ABC system due to its complexity (e.g., missing the opportunity to improve pricing decisions because of increased accuracy of product costs). Reduction of these two types of costs will automatically follow from any reduction in cost system complexity.

The purpose of this research is to explore *after-the-fact* simplification and establish the existence of parsimonious costs systems—those that reduce the complexity and cost of an existing activity-based system without reducing its accuracy. We show that parsimonious cost systems do exist to the extent that a set of linearly independent consumption vectors exists. Through linear independence we show that these smaller systems exist, with the number of cost drivers equal to at most the number of products in the firm. We also show that the number of cost drivers can be reduced even further using the properties of vectors and vector subspaces, while still maintaining the desired level of accuracy.

The focus of this research is cost system simplification for a decision-making context and as such we do not address the issues of control or strategic management. This study is a significant extension of prior studies where cost system simplification was achieved but with an accepted loss in product costing accuracy (Babad & Balachandran, 1993; Homburg, 2001). This research is important for providing information for decisions that rely on accurate product costs, such as pricing, target costing, and relevant costing.

In the next section, the relevant prior literature is discussed. Then a definition of a complex ABC system is presented and is used to illustrate how parsimonious cost systems can be identified using the properties of linear independence. A discussion of the economic and practical implications of parsimonious systems follows, and conclusions are summarized in the final section.

Literature Review and Motivation

The accuracy of product costing information is important for effective decisions related to products. Compared to traditional volume-based product costing systems, ABC is argued to be more accurate (Babad & Balachandran, 1993). This purported accuracy stems from the identification of the causal relationships between costs and their drivers, which increases the number of cost pools and drivers compared to a traditional volume-based system. However, Datar and Gupta (1994) show that increasing the number of cost pools and cost drivers, as ABC requires, *may not* increase product-costing accuracy because of the unfavorable tradeoffs that can occur with aggregation, specification, and measurement errors.[1] They note that: "In theory, costing systems can be designed such that specification and aggregation errors will be minimal" (Datar & Gupta, 1994, 585).

Knowing that no costing system is free from these errors, Labro and Vanhoucke (2008) find that robustness to measurement and specification error tends to increase as the number of driver links decreases. The results of their study suggest that decreasing the number of drivers actually results in a system that is more robust to measurement and specification errors.

Babad and Balachandran (1993) and Homburg (2001) present approaches that specifically address the issue of the number of drivers and cost, noting that there is a trade-off between accuracy and the cost and complexity of a cost system. In fact, Homburg (2001) claims that reducing the possible cost drivers always reduces accuracy. Thus, the two studies propose models that seek a balance between accuracy benefits and the ongoing complexity and costs of data collection, storage, and processing associated with a completely specified ABC system.

Babad and Balachandran (1993) begin with a fully specified ABC driver set and develop a model that identifies an optimal subset of drivers that takes into consideration information production costs and accuracy. The optimal subset of drivers is selected by maximizing the difference between the information cost savings of the eliminated drivers and the costs of lost accuracy. The model allows the decision maker to specify the maximum number of drivers allowed in the simplified system (as a constraint). Error measures are calculated for the simplified system. The approach combines the costs of the activities corresponding to the eliminated drivers with the activity costs associated with the selected drivers, defining a new (more aggregate) cost pool for each selected driver. In building these more aggregate cost pools, all of the associated activity costs of an eliminated driver are transferred to the cost pool of a corresponding selected driver.

Homburg (2001) extends the Babad and Balachandran model by allowing the activity costs of the eliminated drivers to be allocated to multiple selected drivers, rather than just one. The optimal subset of drivers is selected that minimizes accuracy loss with information costs expressed as a constraint in the model (drivers are selected that do not exceed a prespecified level of information production costs). Other constraints ensure that the costs of eliminated activities are allocated among the surviving activities. Thus, the cost pool for a selected driver is the cost of the selected driver's associated activity plus a share of the costs of the eliminated activities. He then shows his approach creates a simplified system with the same ABC-system complexity as the Babad and Balachandran approach but with more accurate product costs.

The fact that Homburg's model produces a more accurate system with no greater information production cost illustrates that the Babad and Balachandran model did not identify the optimal simplified system. However, both models assume a simplified system must sacrifice accuracy. Datar and Gupta (1994) also make this assumption because their analysis implies that moving from a system that has minimized the three types of error to a simplified system must necessarily increase aggregation and specification error. This article illustrates that the accuracy-reduction assumption is not valid because it is indeed possible to reduce the number of drivers and simultaneously aggregate costs in such a way that minimal accuracy is sacrificed relative to a fully specified ABC system.

Parsimonious Systems with Reduced Complexity

The Benchmark ABC System

To establish the existence of parsimonious systems, it is first necessary to define an existing complex ABC system. In theory, an ABC system identifies as many activities as possible, determines accurately the cost of each activity, and then uses a properly specified, unique activity driver to assign the activity cost to products. For simplification, we ignore the assignment of direct costs. These costs would be assigned in the same manner regardless of the system used to assign indirect costs.

The amount of activity cost assigned to a product is proportional to the amount of the activity driver consumed by the product. However, if two drivers are perfectly correlated, then only one of these is used to assign the combined cost of the two activities (Babad & Balachandran, 1993). After this adjustment, we assume that there remain s activities. Assuming s mutually exclusive activities and n products, the total amount of indirect cost assigned to each product is simply the sum of the cost received from each activity:

$$\alpha_i = \sum_{j=1}^{s} \delta_{ij} c_j^o, \qquad i = 1, \ldots, n, \tag{1}$$

where:

α_i = The expected activity-based indirect product cost;
δ_{ij} = The expected proportion of the cost of activity j assigned to product i (i.e., the consumption ratio);
c_j^o = The cost of activity j.

The expected ABC indirect product cost assignment for product i, α_i, is the benchmark for evaluating the accuracy of simplified costing systems. Letting c = the total overhead costs, an alternative and equivalent benchmark is the expected global consumption ratio, ε_i, which is defined as follows:

$$\varepsilon_i = \frac{\alpha_i}{c} = \sum_{j=1}^{s} \left(\frac{c_j^o}{c}\right) \delta_{ij}. \tag{2}$$

The expected global consumption ratio is the weighted average of the individual consumption ratios. The expected consumption ratios, δ_{ij}, are assumed to be stable for a minimum of t periods, $t \geq 1$.

Less Complex Systems

A less complex system is one with fewer drivers than the number in the benchmark system. Reducing the number of drivers requires that the total overhead costs be assigned to the remaining drivers in the less complex system. The critical question is whether the cost pools for the reduced set of drivers can be defined such that the associated drivers provide the same cost assignment as the benchmark ABC assignment. Using the concepts of linearly independent and dependent vectors, it can be shown that it is possible to simplify the ABC system without sacrificing accuracy. Every product consumes an expected proportion of an activity, as measured by its associated driver. Thus, for activity j, there is an n-dimensional consumption ratio vector (usually expressed as an $n \times 1$ column vector), $\delta_j = (\delta_{1j}, \delta_{2j}, ..., \delta_{nj})'$.[2] For the benchmark ABC set of s consumption ratio vectors, there are at most n linearly independent vectors. Under the assumption that the number of products is less than the number of activities (which holds for many, if not most settings), the set of s consumption ratio vectors is a dependent set. All dependent activities within the benchmark set can be expressed as a linear combination of the independent activities and, therefore, can be eliminated. The costs of the eliminated dependent activities are reassigned to the independent activities using the scalars that define the relationship between the dependent and independent activities. This approach creates an accuracy equivalent simplified system as stated in the following proposition:

Proposition 1. Assume $s > n$. If there exist m linearly independent consumption ratio vectors, $1 < m \leq n$, then a simplified system of m drivers exists that matches ABC indirect cost assignments.

Proof. Let c_j^o = the cost of activity j in the benchmark ABC system and δ_j = the vector of consumption ratios for activity j, where δ_{ij} is the ith component of the vector. Number the s activities so that the first m is linearly independent. Since all vectors, except for the first m, can be expressed as a linear combination of the m linearly independent vectors, we have the following: $\sum_{j=1}^{m} k_{rj}\delta_j = \delta_r$, $r = m + 1, \ldots, s$. Using the scalars, k_{rj}, as allocation rates, define the cost assigned to activity j, $j = 1, \ldots, m$, as $c_j = \sum_{r=m+1}^{s} k_{rj}c_r^o + c_j^o$. Thus, the total indirect cost, α_i^m, assigned to product i in the m-driver system is expressed as:

$$\alpha_i^m = \sum_{j=1}^{m} \delta_{ij} c_j$$

$$= \sum_{j=1}^{m} \delta_{ij} [\sum_{r=m+1}^{s} k_{rj} c_r^o + c_j^o]$$

$$= \sum_{j=1}^{m} \delta_{ij} c_j^o + \sum_{j=1}^{m} \delta_{ij} \sum_{r=m+1}^{s} k_{rj} c_r^o$$

$$= \sum_{j=1}^{m} \delta_{ij} c_j^o + \sum_{j=m+1}^{s} \delta_{ij} c_j^o = \alpha_i \qquad \square.$$

Babad and Balachandran (1993) have shown that when two drivers are perfectly correlated, then the two drivers can be combined without loss of accuracy. This is equivalent to stating that when two drivers have exactly the same expected consumption ratio vector, then the costs of the two activities can be added and the combined costs assigned using either driver (since each driver assigns the costs in the same proportion). As stated, this reduces the number of drivers needed to a total of s. Proposition 1 generalizes this finding from Babad and Balachandran's Lemma 1. Without loss of accuracy, it establishes that it is possible to reduce the ABC system from s to at most n drivers. Expressing a driver's consumption ratio vector as a linear combination of m linearly independent vectors is essentially stating that the dependent driver is perfectly correlated with the m independent drivers. Thus, all dependent activities can be combined with the independent activities. The combination of a dependent activity with m activities is more complicated than combining with a single activity because the costs of the dependent activities must be assigned in a very specific way. This assignment not only allocates the dependent activity's costs to the m activities, it may also redistribute some of the cost of an independent activity to other independent activities. For example, the scalar coefficients, which act as allocation rates, can be zero, positive, or negative.

Example of Simplification with Linearly Independent Consumption Ratio Vectors

TABLE 1 provides example product costing data for indirect costs, the ABC cost assignment, the global consumption ratios for a completely specified benchmark ABC system, and the indirect product cost assignment using a traditional, plant-wide rate based on direct labor hours. Of the ten drivers, seven through nine are unit-level drivers with the other seven classified as non-unit level drivers. Letting ϕ_i be the total cost of product i using a plant-wide rate based on direct labor hours, the assigned indirect product costs would be ϕ_1 = \$500, ϕ_2 = \$1,500, ϕ_3= \$2,000, and ϕ_4 = \$6,000. Comparing these costs to the benchmark ABC cost assignment, it is evident that the plant-wide rate significantly distorts product costs. Thus, a traditional system based on direct labor hours is not a desirable simplification.

TABLE 1 Product costing data—Indirect costs: Benchmark ABC System

	Activity Driver[a]												
Product	Parts (1)	Moves (2)	Eng. Hours (3)	Batches (4)	Setup Hours (5)	Orders (6)	Materials (7)	Direct Labor Hours (8)	Machine Hours (9)	Invoices (10)	α^b: ABC Cost Assignment	ε^c: Global Ratios (ABC)	ϕ^c: Traditional Cost Assignment
P1	0.00	0.00	0.125	0.15	0.20	0.10	0.18	0.05	0.075	0.12	$1,000	0.10	$500
P2	0.00	0.25	0.295	0.15	0.15	0.30	0.19	0.15	0.225	0.29	$2,000	0.20	$1,500
P3	0.25	0.25	0.230	0.45	0.35	0.25	0.37	0.20	0.400	0.25	$3,000	0.30	$2,000
P4	0.75	0.50	0.350	0.25	0.30	0.35	0.26	0.60	0.300	0.34	$4,000	0.40	$6,000
Activity cost	$1,556	$544	$1,000	$675	$1,400	$2,100	$500	$200	$1,300	$725	$10,000	1.00	$10,000

[a]Activity drivers are numbered (1) through (10) and have the indicated labels. The consumption ratios for each driver are provided in rows labeled P1 through P4.
[b]Multiplying the consumption ratio for each product by the corresponding activity cost and then summing across activities yields the activity-based overhead cost assignments. Costs are rounded to the nearest hundred.
[c]The global ratios are computed by dividing each activity-based cost assignment by the total overhead cost of $10,000 or by using the following formula: $\boldsymbol{\varepsilon}_i = \sum_{j=1}^{s} \left(\frac{c_j^o}{c} \right) \boldsymbol{\delta}_{ij}$. The ratios are rounded to the hundredth decimal place.
[d]The traditional overhead cost assignment is obtained by simply multiplying the direct labor consumption ratio for each product by the total overhead cost of $10,000.

Proposition 1 indicates that a simplified system with at most 4 drivers will produce the same accuracy as the 10-driver system. Of course, the need for at most 4 drivers is true even if the example were expanded to have 400 drivers. It is easy to verify that the first four consumption ratio vectors, $\boldsymbol{\delta}_j, j = 1, \ldots, 4$, are linearly independent.[3] Thus, the remaining six drivers can be expressed as a linear combination of drivers 1 to 4:

$$\sum_{j=1}^{4} k_{rj}\boldsymbol{\delta}_j = \boldsymbol{\delta}_r, \qquad r = 5, 6, 7, 8, 9, 10. \tag{3}$$

Equation (3) produces six sets of simultaneous equations. For example, the set of simultaneous equations corresponding to driver 5 is expressed as follows:

$$k_{51}\begin{bmatrix} 0.00 \\ 0.00 \\ 0.25 \\ 0.75 \end{bmatrix} + k_{52}\begin{bmatrix} 0.00 \\ 0.25 \\ 0.25 \\ 0.50 \end{bmatrix} + k_{53}\begin{bmatrix} 0.125 \\ 0.295 \\ 0.230 \\ 0.350 \end{bmatrix} + k_{54}\begin{bmatrix} 0.15 \\ 0.15 \\ 0.45 \\ 0.25 \end{bmatrix} = \begin{bmatrix} 0.20 \\ 0.15 \\ 0.35 \\ 0.30 \end{bmatrix} \tag{4}$$

The other five sets of equations are similar with the right hand side vector being changed to the corresponding dependent activity. Solutions to the six sets of equations are provided in TABLE 2. Note that the k_{rj}'s sum to one for every dependent activity. This is always the case as can be seen by summing the four equations represented by Equation (4). Also notice that several of the k_{rj}'*s* are negative. For example, a negative k_{82} means that all of activity 8's costs and some of activity 2's costs are assigned to activities 1, 3, and 4. A similar interpretation is given for the other negative k_{rj}.

Using the allocation rates of TABLE 2, the costs assigned to the four independent activities are calculated using the following formula:

$$c_j = \sum_{r=5}^{9} k_{rj}c_r^o + c_j^o, \qquad j = 1, ..., 4. \tag{5}$$

TABLE 2 Scalar coefficients[a] (allocation rates) for each dependent activity (from Table 1)

Independent Activities	Allocation Rates for Dependent Activities[b]					
	k_{5j}	k_{6j}	k_{7j}	k_{8j}	k_{9j}	k_{10j}
$j = 1$	0.325	−0.100	0.090	0.588	−0.237	−0.040
$j = 2$	−0.807	0.314	−0.397	−0.025	0.782	0.097
$j = 3$	0.893	0.715	0.643	0.625	−0.268	0.857
$j = 4$	0.589	0.071	0.664	−0.188	0.723	0.086
$\sum k_{rj}$	1.000	1.000	1.000	1.000	1.000	1.000

[a]The scalar coefficients, k_{rj}, represent allocation rates used to allocate the cost of the dependent activities to the remaining independent activities.
[b]The k_{rj}'s presented in the table are the result of solving the following sets of simultaneous equations for the corresponding data in Table 1: $\sum_{j=1}^{4} k_{rj}\boldsymbol{\delta}_j = \boldsymbol{\delta}_r$; $r = 5, \ldots, 10$, where $\boldsymbol{\delta}_j$ = the consumption ratio vector for activity j and $\boldsymbol{\delta}_r$ = the consumption ratio vector for activity r.

TABLE 3 Parsimonious four-driver system[a]

Product	Activity Drivers: Parts (1)	Moves (2)	Eng. Hours (3)	Batches (4)	α: Simplified ABC Indirect Cost Assignment	ε: Global Ratios
P1	0.00	0.00	0.125	0.15	\$1,000	0.10
P2	0.00	0.25	0.295	0.15	\$2,000	0.20
P3	0.25	0.25	0.230	0.45	\$3,000	0.30
P4	0.75	0.50	0.350	0.25	\$4,000	0.40
Cost[b] pool	\$1,625.75	\$957.64	\$4,469.64	\$2,946.96	\$10,000	

[a]A simplification of the ten driver system presented in Table 1.
[b]The cost pools are the result of applying the allocation rates presented in Table 2 in the following manner: $c_j = \sum_{r=5}^{10} k_{rj} c_r^0 + c_j^0$, $j = 1, \ldots, 4$, where k_{rj} = cost allocation rate for allocating the cost of the dependent activities to the independent activities, c_r^0 = initial cost of activity r, and c_j^0 = initial cost of activity j.

The simplified systems corresponding to those displayed in TABLE 1 are provided in TABLE 3.

Parsimonious Systems with r < n

Proposition 1 establishes the fact that simpler, equally accurate systems are possible. We know this can be done because n linearly independent drivers will span R^n and $\boldsymbol{\varepsilon}$ (the global consumption ratio vector) is a member of R^n. Thus, for a system that has hundreds of activities, there are at most n linearly independent consumption ratio vectors. Assuming that n is much smaller than the actual number of activities, then we have achieved a significant simplification in complexity. However, n can still be large—much larger than what would be typically found in a traditional costing system. This poses the interesting question of whether or not systems exist that can match ABC accuracy with fewer drivers than n, even when there are n linearly independent consumption ratio vectors.

Parsimonious systems smaller than n are also possible. This can be shown using the concepts of vector spaces and vector subspaces. Let V be a set of vectors (R^n, for example). V is a vector space if two conditions are met: (1) for every pair of vectors in V, their sum is also in V, and (2) the product of a scalar and any vector in V is also in V. A vector subspace, W, is a vector space that is contained in V ($W \subseteq V$). If $W \subset V$, then W is a proper subspace of V. The dimension of a vector space is the number of linearly independent vectors needed to span the space. A proper subspace, W, can have a dimension less than V. For example, the two linearly independent consumption ratio vectors, (0.5, 0.5, 0, 0) and (0, 0, 0.5, 0.5), span all consumption ratio vectors of the form (a, a, b, b). This subspace thus has a dimension of 2. These two linearly independent consumption ratio vectors would also span the global consumption ratio vector if it were of the form (a, a, b, b), as would four linearly independent vectors. Thus, smaller systems exist whenever there are vector subspaces of consumption ratios that (1) have a dimension $r < n$, and (2) the dimension of the subspace remains unchanged when $\boldsymbol{\varepsilon}$ is added to the set.

Spanning Subspace Vectors. Spanning subspace vectors of dimension $r < n$ consist of r linearly independent consumption ratio vectors that span the global consumption ratio vector.

For s activities, the candidates for a spanning subspace total ${}_sC_r$ where ${}_sC_r$ = the number of combinations of s drivers taken r at a time. Those combinations where the r vectors are not linearly independent can be immediately eliminated. While there is no guarantee that spanning subspaces will exist for a given setting, it is possible to identify all the potential spanning vectors of dimension r.

Specifically, we are looking for all sets of r linearly independent consumption ratio vectors, $2 \leq r < n$, such that $\sum_{j=1}^{r} w_j \boldsymbol{\delta}_j = \varepsilon$ where $w_j > 0$ and $\sum_{j=1}^{r} w_j = 1$. Consider the set, K, of all linearly independent consumption ratio vectors of dimension r - 1. Let $\boldsymbol{\delta}^k = (\boldsymbol{\delta}_1^k, \boldsymbol{\delta}_2^k, \ldots, \boldsymbol{\delta}_{r-1}^k)$, where $\boldsymbol{\delta}^k \in K$ and $k = 1, 2, \ldots, p$ with $p \leq {}_sC_{r\text{-}1}$. Let $\boldsymbol{\delta}_r^{cv}$ be the rth consumption ratio vector, a completing vector. Then $\boldsymbol{\delta}_r^{cv}$ added to $\boldsymbol{\delta}^k$ defines a spanning subspace vector based on the following:

$$w_r \boldsymbol{\delta}_r^{cv} + \sum_{j=1}^{r-1} w_j \boldsymbol{\delta}_j^k = \boldsymbol{\varepsilon}, \tag{6}$$

$$\boldsymbol{\delta}_r^{cv} = \frac{\boldsymbol{\varepsilon} - \sum_{j=1}^{r-1} w_j \boldsymbol{\delta}_j^k}{1 - \sum_{j=1}^{r-1} w_j}, \tag{7}$$

$$w_j \leq \min_i \left(\frac{\boldsymbol{\varepsilon}_i}{\boldsymbol{\delta}_{ij}} \right), \qquad j = 1, 2, \ldots r - 1; \qquad i = 1, 2, \ldots, n, \tag{8}$$

$$\sum_{j=1}^{r-1} w_j < 1. \tag{9}$$

Equation (7) ensures that the r vectors, $(\boldsymbol{\delta}_1^k, \boldsymbol{\delta}_2^k, \ldots, \boldsymbol{\delta}_r^{cv})$, span the global consumption ratio vector while Equations (8) and (9) ensure that the chosen w_j's produce a positive $\boldsymbol{\delta}_r^{cv}$. Allowing the w_j, $j = 1, \ldots, r$ - 1, to vary within the permitted ranges generates all vectors, $\boldsymbol{\delta}_r^{cv}$, which, when added to the r - 1 vectors, will define spanning subspace vectors. Repeating this process for every $\boldsymbol{\delta}^k \in K$ generates all potential spanning subspace vectors. Letting M_r be the set of all completing vectors for K, we state the following proposition that follows from the above arguments:

Proposition 2. If one of the firm's consumption ratio vectors belongs to the completing set, M_r, then a parsimonious system exists of size r.

Examples of Reduced Systems Using Subspace Vectors. Using the data of TABLE 1, it is possible to show that subspace vectors exist so that parsimonious systems with $r = 2$ and $r = 3$ can be defined. For $r = 2$, K is simply all 10 activities. Consider $\boldsymbol{\delta}_1^2 \in K$ (driver 2 of TABLE 1). Equations (8) and (9) imply that $w_1 \leq 0.8$. Thus, Equation (7) and any $w_1 \in [0, 0.8]$ produce a completing vector for $\boldsymbol{\delta}_1^2$. For example, applying Equation (7) to each w_1 obtained by setting $w_1 = 0.1$ and allowing it to increase by increments of 0.1 produces the eight completing vectors shown in TABLE 4. Comparing the completing vectors of TABLE 4 with the consumption ratio vectors in TABLE 1, we see that the completing vector with a weight of 0.5 corresponds to driver 5 of TABLE 1; therefore, consistent with Proposition 2,

TABLE 4 Completing vectors for a two-driver system

	Weights for Completing Vectors[a]							
δ_1^2	$w_2 = 0.9$	$w_2 = 0.8$	$w_2 = 0.7$	$w_2 = 0.6$	$w_2 = 0.5$	$w_2 = 0.4$	$w_2 = 0.3$	$w_2 = 0.2$
0.00	0.111	0.125	0.143	0.167	0.200	0.250	0.333	0.50
0.25	0.194	0.188	0.179	0.167	0.150	0.125	0.083	0.00
0.25	0.306	0.312	0.321	0.333	0.350	0.375	0.417	0.50
0.50	0.389	0.375	0.357	0.333	0.300	0.250	0.167	0.00

[a] $w_1 \in$ **[0.0.8]** is required for any consumption ratio vector to complete a two-dimensional vector subspace that spans the global consumption ratio vector. The weight (allocation rate) for the completing vector is $w_2 = 1 - w_1$. The completing vectors are calculated for each of the above weights using the following formula: $\delta_2^{cv} = \frac{\varepsilon - w_1 \delta_1^2}{1 - w_1}$, where $\delta_2^c = a$ completing vector for δ_1^2.

a system with two drivers exists that replaces the completely specified ABC system and has the same degree of accuracy.

Thus, drivers 2 and 5 using weights $w_1 = 0.5$ and $w_2 = 0.5$ span the global vector:

$$0.5 \begin{bmatrix} 0.00 \\ 0.25 \\ 0.25 \\ 0.50 \end{bmatrix} + 0.5 \begin{bmatrix} 0.20 \\ 0.15 \\ 0.35 \\ 0.30 \end{bmatrix} = \begin{bmatrix} 0.10 \\ 0.20 \\ 0.30 \\ 0.40 \end{bmatrix} \tag{10}$$

Since the weights are allocation rates, the cost pools assigned to the two drivers are $c_2 = 0.5 \times \$10{,}000 = \$5{,}000$ and $c_5 = 0.5 \times \$10{,}000 = \$5{,}000$. This two-driver system is summarized in TABLE 5.

For three-driver systems, we must identify all completing vectors for all combinations of two vectors. There are 45 pairs (${}_{10}C_2$) to evaluate. For example, consider the kth pair: drivers 2 and 8 of TABLE 1. Letting w_1 be the allocation rate for driver 2 and w_2 the allocation rate for driver 8, all (w_1, w_2) pairs that satisfy Equations (8) and (9) ($w_1 \leq 0.8$, $w_2 \leq 0.67$, and $w_1 + w_2 < 1$) define the set of completing vectors. For example, if $w_1 = 0.3$ and $w_2 = 0.2$, then the completing vector calculated by Equation (7) corresponds to driver 7 of TABLE 1. Thus, drivers 2, 7, and 8 with allocation weights of 0.3, 0.2, and 0.5 span the global vector:

TABLE 5 Parsimonious two-driver system[a]

	Activity Drivers			
Product	Moves (2) (δ_1^2)	Setup Hours (5) (δ_2^{cv})	α: Simplified ABC Indirect Cost Assignment	ε: Global Ratios
P1	0.00	0.20	\$1,000	0.10
P2	0.25	0.15	\$2,000	0.20
P3	0.25	0.35	\$3,000	0.30
P4	0.50	0.30	\$4,000	0.40
Cost[b]	\$5,000	\$5,000	\$10,000	

[a] A simplification of the ten-driver system in Table 1.
[b] The cost assigned to each of the activities is based on the weights, $w_1 = 0.5$ and $w_2 = 0.5$.

$$0.3\begin{bmatrix}0.00\\0.25\\0.25\\0.50\end{bmatrix}+0.2\begin{bmatrix}0.05\\0.15\\0.20\\0.60\end{bmatrix}+0.5\begin{bmatrix}0.18\\0.19\\0.37\\0.26\end{bmatrix}=\begin{bmatrix}0.10\\0.20\\0.30\\0.40\end{bmatrix} \quad (11)$$

A parsimonious system is possible using only three drivers, and the costs assigned to activities 2, 7, and 8 are $c_2 = 0.3 \times \$10{,}000 = \$3{,}000$, $c_7 = 0.5 \times \$10{,}000 = \$5{,}000$, and $c_8 = 0.2 \times \$10{,}000 = \$2{,}000$. This three-driver parsimonious system is summarized in TABLE 6.

Identifying All Parsimonious Systems, $r \leq n$

Thus far, we have provided examples of parsimonious systems. A systematic procedure for exhaustively identifying all such systems would be useful. From Proposition 1, we know that we can solve for the cost pools of n linearly independent consumption ratio vectors using the following equation:

$$\sum_{j=1}^{n} c_j \boldsymbol{\delta}_j = \boldsymbol{\alpha}. \quad (12)$$

Dividing both sides of Equation (12) by c, the total overhead cost, we obtain an equivalent equation:

$$\sum_{j=1}^{n} w_j \boldsymbol{\delta}_j = \boldsymbol{\varepsilon}, \quad (13)$$

where $w_j = c_j/c$. In matrix notation, Equation (13) is expressed as:

$$D\mathbf{w} = \boldsymbol{\varepsilon}, \quad (14)$$

TABLE 6 Parsimonious three-driver system[a]

	Activity Drivers[b]			$\boldsymbol{\alpha}$:	$\boldsymbol{\varepsilon}$:
Product	Moves (2) ($\boldsymbol{\delta}_1^k$)	Setup Hours (8) ($\boldsymbol{\delta}_2^k$)	Materials (7) ($\boldsymbol{\delta}_3^{cv}$)	Simplified ABC Indirect Cost Assignment	Global Ratios
P1	0.00	0.05	0.18	\$1,000	0.10
P2	0.25	0.15	0.19	\$2,000	0.20
P3	0.25	0.20	0.37	\$3,000	0.30
P4	0.50	0.60	0.26	\$4,000	0.40
Cost[c]	\$3,000	\$2,000	\$5,000	\$10,000	

[a]A simplification of the ten driver system presented in Table 1.
[b]We let the two-driver combination of 2 and 8 be combination k.
[c]The cost assigned to each of the three activities is based on the weights, $w_1 = 0.3$, $w_2 = 0.2$, and $w_3 = 0.5$. Multiplying these weights by the total overhead cost of \$10,000 results in the three cost pools for each driver.

where D is an $n \times n$ matrix of consumption ratios and $\mathbf{w}$ and $\boldsymbol{\varepsilon}$ are $n \times 1$ column vectors. Solving Equation (14) for w, we obtain:

$$\mathbf{w} = \mathbf{D}^{-1}\boldsymbol{\varepsilon} \tag{15}$$

All those systems with positive cost pools ($\boldsymbol{w} \geq \mathbf{0}$, with $w_i > 0$ for at least one i), qualify as parsimonious systems. Subspaces are identified whenever a $\boldsymbol{w}$ has one or more $w_i = 0$.

If all combinations of size n (${}_sC_n$ = the number of n-combinations) were linearly independent, then solving Equation (14) would exhaustively identify all parsimonious systems with positive cost pools. Attempting to solve Equation (14) will segment the n-combinations into two subsets: those where the inverse matrix D^{-1} exists (the set of linearly independent n-combinations) and those where it does not exist (the set of all linearly dependent n-combinations). For each linearly dependent n-combination, it is possible that one of its sub-combinations will span the global vector. However, if the sub-combinations are found in other independent n-combinations, then the evaluation of these sub-combinations has already been done. For example, consider a four-product system and suppose that drivers 3, 4, 5, and 6 form a dependent combination. There are four three-driver sub-combinations (3-4-5, 3-4-6, 3-5-6, and 4-5-6) and six two-driver combinations (3-4, 3-5, 3-6, 4-5, 4-6, and 5-6). Now assume that if driver 1 is added to each of the four three-driver combinations, a linearly independent combination is produced. In this case, the sub-combinations have been evaluated in this independent set and no further action is needed (the two-driver combinations are also included in the independent combinations). Finally, suppose that sub-combination 3-4-5 is not found in an independent four-driver combination. The combination can be evaluated by adding a unitary product vector, $\boldsymbol{e}_i$, so that the combination $\boldsymbol{e}_i$-3-4-5 is linearly independent.

Example Illustrating Identification Approach

Using the data of TABLE 1, we identify all parsimonious systems for $r = 2$, 3, and 4. There are 210 combinations of size four (${}_{10}C_4$). Solving Equation (14) for these 210 combinations produces 208 linearly independent combinations and two linearly dependent combinations (1-6-8-9 and 6-7-9-10). Since all sub-combinations of the dependent combinations are evaluated in one of the linearly independent combinations, no additional action is needed. Of the 208 linearly independent combinations, there were 27 different combinations of four drivers that replaced ABC assignments with positive allocation ratios, two different duplicating combinations with three drivers, and one combination with two drivers. These combinations and their associated allocation weights are shown in TABLE 7.

Parsimonious Systems with Reduced Cost

Having established that reducing complexity without reducing accuracy is possible, the next objective is to determine whether the ABC system cost can be reduced without sacrificing accuracy. Three categories of costs associated with a traditional ABC system will be considered: (1) complexity costs; (2) ongoing costs (data collection and updating); and (3) implementation costs. Complexity costs are the costs of foregone benefits because managers reject the ABC system due to its complexity (e.g., missing the opportunity to improve pricing decisions because of increased accuracy of product costs). Ongoing data collection costs are the costs incurred to measure expected and actual consumption of activities so that applied overhead costs can be computed (predetermined activity rates $\times$ the actual

TABLE 7 Parsimonious systems (with cost allocation weights for the associated drivers)[a]

Drivers	Allocation Weights[b]	Drivers	Allocation Weights[b]	Drivers	Allocation Weights[b]
2, 5	0.50, 0.50	1, 3, 5, 9	0.18, 0.39, 0.15, 0.28	1, 7, 9, 10	0.20, 0.20, 0.17, 0.43
2, 7, 8	0.30, 0.50, 0.20	1, 3, 7, 9	0.20, 0.38, 0.20, 0.22	2, 3, 4, 8	0.10, 0.27, 0.35, 0.28
4, 6, 8	0.33, 0.33, 0.34	1, 4, 5, 6	0.19, 0.21, 0.08, 0.52	2, 4, 8, 10	0.08, 0.32, 0.30, 0.30
1, 2, 3, 4	0.16, 0.10, 0.29, 0.45	1, 4, 6, 7	0.20, 0.16, 0.49, 0.15	3, 4, 8, 9	0.28, 0.26, 0.33, 0.13
1, 2, 4, 10	0.18, 0.05, 0.25, 0.52	1, 4, 6, 10	0.20, 0.25, 0.20, 0.35	3, 5, 8, 9	0.17, 0.19, 0.31, 0.33
1, 2, 6, 7	0.15, 0.18, 0.25, 0.42	1, 4, 9, 10	0.20, 0.21, 0.05, 0.24	3, 7, 8, 9	0.14, 0.25, 0.35, 0.26
1, 2, 3, 7	0.12, 0.27, 0.45, 0.16	1, 5, 6, 9	0.18, 0.22, 0.42, 0.18	5, 6, 8, 9	0.22, 0.19, 0.30, 0.29
1, 2, 7, 10	0.13, 0.24, 0.42, 0.21	1, 5, 8, 9	0.18, 0.15, 0.39, 0.28	5, 8, 9, 10	0.19, 0.31, 0.31, 0.19
1, 3, 4, 6	0.19, 0.23, 0.27, 0.31	1, 5, 9, 10	0.18, 0.16, 0.23, 0.43	6, 7, 8, 9	0.15, 0.28, 0.35, 0.22
1, 3, 4, 9	0.19, 0.48, 0.21, 0.12	1, 6, 7, 9	0.20, 0.41, 0.29, 0.10	7, 8, 9, 10	0.26,0.35,0.24,0.15

[a] Representing the ten-driver system presented in Table 1.

[b] Cost allocation weights are a result of solving $D\mathbf{w} = \boldsymbol{\varepsilon}$, where D is an $n \times n$ matrix of consumption ratios, and $\mathbf{w}$ and $\boldsymbol{\varepsilon}$ are $n \times 1$ column vectors of allocation weights and global consumption ratios, respectively. There is an error present due to rounding.

amount of activity consumed). Thus, activity costs and expected consumption patterns for each activity must be estimated and data must be collected for each driver so that overhead costs can be assigned to products as production unfolds. The more drivers there are, the more data collection needed. Updating costs are the costs of adjustments that must be made for significant changes in underlying relationships. Implementation costs are the initial costs of identifying activities and drivers and assigning the costs of resources to activities (first-stage allocation).

Reducing Complexity and Ongoing Costs

Reducing complexity decreases the likelihood that managers will reject an ABC system and this in turn reduces complexity costs. Reducing complexity also implies that the ongoing data collection costs can be reduced because there are fewer drivers needed to apply overhead costs to products. In fact, ongoing data collection costs can be totally avoided for at least k periods by simply using $c\boldsymbol{\varepsilon}$ to calculate the expected overhead costs for each product.[4] Using $\boldsymbol{\varepsilon}$ to assign overhead costs in this way is an example of a particular n-dimensional parsimonious cost system. However, $c\varepsilon$ is the total expected overhead cost for each product. Overhead is applied on a day-by-day basis by dividing $c\varepsilon$ by the expected units for each product to obtain an expected overhead cost per unit. Thus, there is an implicit driver: units of *each* product. Since every cost system gathers this particular driver information, there is no incremental ongoing data collection required.

Other parsimonious systems are also possible—even smaller than dimension n if desired—where the more familiar practice of using a small number of common drivers to assign overhead costs may be used. Furthermore, following the familiar practice of applying overhead costs using activity drivers rather than the weighted average of all drivers may be a better approach if the objective is to gain managerial acceptance of an activity-based system. Although using drivers to assign overhead costs increases ongoing data collection costs relative to the use of $\boldsymbol{\varepsilon}$ and units of product, because $r < s$ it still significantly reduces the data collection costs relative to the completely specified system. Thus, we can conclude that complexity costs and ongoing data collection costs can be significantly reduced for parsimonious systems.

Updating costs are another matter. If $\boldsymbol{\varepsilon}$ has not changed (e.g., no products added or dropped or no major technological changes), then no updating costs are incurred. If $\boldsymbol{\varepsilon}$ has changed, then updating costs must be incurred to assess the new values. Since parsimonious systems depend on the new $\boldsymbol{\varepsilon}$, updating costs are essentially identical for both the completely specified system and the parsimonious systems. However, since updating activities are essentially a smaller-in-scope repetition of implementation activities, updating costs can be reduced if implementation costs can be reduced.

Reducing Implementation Costs

Reducing the Size of the Benchmark System. Parsimonious systems require knowledge of $\boldsymbol{\varepsilon}$, implying that the implementation costs of a completely specified benchmark ABC system must be incurred. Using a smaller, well-specified benchmark system that estimates $\boldsymbol{\varepsilon}$ with an acceptable level of confidence and precision would be one way of reducing implementation and updating costs (by reducing the number of times an implementation activity is performed). Sampling techniques may be useful for decreasing the size of the benchmark system. A random sample of activities can produce very good estimates of the global consumption ratios, where the sample size is significantly smaller than the total number of activities. Reducing implementation costs in this way would thus facilitate the viability of establishing and maintaining parsimonious cost systems.

Reducing First-Stage Allocation Costs. Another possibility is to reduce implementation costs by eliminating the need for first-stage allocation. Eliminating the need to identify and assign the costs of resources to individual activities would produce significant savings and make it much more practical to assess and maintain knowledge of $\boldsymbol{\varepsilon}$. It is possible to eliminate or reduce first-stage allocations depending on the nature of the correlation between consumption ratios and activity costs. Since there is no reason to believe that the consumption ratio of product i must increase (decrease) as we move from a less (more) costly activity to a more (less) costly activity, an assumption of no correlation appears the most reasonable. Under this no-correlation assumption, we can state and prove the following:

Proposition 3. If there is no correlation between δ_{ij} and c_j, $i = 1, \ldots, n$ - 1 and $j = 1, \ldots s$, then $\alpha_i = \overline{\delta}_i c$, $i = 1, \ldots, n$ - 1, and there is no need to know individual activity costs.

Proof.

$$Cov(\delta_{ij}, c_j) = 0 \Rightarrow E(\delta_{ij} - \overline{\delta_i})(c_j - \bar{c}) = 0 \Rightarrow E(\delta_{ij}c_j) - \overline{\delta_i}\bar{c} = 0 \Rightarrow E(\delta_{ij}c_j) = \overline{\delta_i}\bar{c}.$$

Since we are dealing with equally weighted discrete values, $E(\delta_{ij}c_j) = \frac{\sum_j \delta_{ij}c_j}{s} = \frac{\alpha_i}{s}$, which implies that $\alpha_i = \overline{\delta_i}c$. Finally, since $\alpha_n = c - \sum_{1=1}^{n-1} \alpha_i$, the activity costs of all products are determined without knowing any c_j.. □

If, according to Proposition 3, there is no correlation for at least n - 1 products, only the total overhead cost, c, and the average consumption ratios are needed to calculate the activity-based product cost for all products. Moreover, since $\alpha_i = \overline{\delta_i}c$ and $\alpha_i = \varepsilon_i c$, we have $\overline{\delta_i} = \varepsilon_i$. In practical terms, the proposition is likely to hold well enough even if there is weak correlation. If there is a strong correlation between δ_{ij} and c_j for more than one product (the no-correlation assumption must hold only for n - 1 products), clearly contradicting Proposition 3, simplification and cost reduction are still possible because a random sample of a reasonably small number of activities allow the relationship to be described by a well-fitted regression equation (requiring, of course, that activity costs be determined for this sample of activities through the usual first-stage allocation methods). This equation is then used to assign the cost to all other activities (outside the sample), avoiding the first-stage allocation costs for these remaining activities.

Examples Illustrating Proposition 3. From TABLE 1, we can calculate the correlation between δ_{ij} and c_j. The largest correlation is 0.002, when $i = 4$. Thus, we can safely assume that the no-correlation assumption holds. The simple average consumption ratios for the four products are 0.1, 0.2, 0.3, and 0.4, equal to the global consumption ratios as predicted, thus allowing the activity-based product costs to be calculated without knowing the individual activity costs (by simply multiplying each $\overline{\delta_i}$ by c, the total overhead cost).

TABLE 8 presents the description of how a larger completely specified system of 400 activities was created. Since activity costs were assigned independently of the consumption ratios, the correlations are not significant, the largest being 0.027 (Product 4). Thus, as expected, $\varepsilon_i = \overline{\delta_i}$. However, if a firm with this structure were considering the

TABLE 8 Four-hundred activity example (completely specified system)

Activities	Uniform Distribution (Activity Cost)[a]	Product	Uniform Distribution (Source of δ_{ij})[b]	$\boldsymbol{\varepsilon}$[c]	$\overline{\boldsymbol{\delta}_i}$[d]
1–100	[\$800, \$1,000]	P1	[0.08, 0.12]	0.10	0.10
101–200	[\$600, \$800]	P2	[0.15, 0.25]	0.20	0.20
201–300	[\$400, \$600]	P3	[0.20, 0.40]	0.30	0.30
301–400	[\$200, \$400]	P4	$1 - \boldsymbol{\delta}_{1j} - \boldsymbol{\delta}_{2j} - \boldsymbol{\delta}_{3j}$	0.40	0.40
Total cost	\$242,554				

[a]Costs are assigned randomly using the indicated uniform distribution.
[b]Expected consumption ratios are assigned randomly to each of the first three products for a given activity using the indicated uniform distributions. The ratios assigned to Product 4 for activity *j* are calculated using the property that the consumption ratios must sum to one.
[c]The weighted average consumption ratio, rounded to the hundredth decimal place.
[d]The simple average consumption ratio, rounded to the hundredth decimal place.

implementation of an ABC system, it would not know up front that the no-correlation assumption holds. One possibility is to simply assume that this assumption is reasonable and press forward. For the more cautious, after identifying all activities and their drivers, a small random sample of activities could be taken and costs assigned so that evidence of correlation can be provided. For the TABLE 8 example, a random sample of 10 activities was taken. TABLE 9 presents the relevant data for this small sample. In this sample, Product 1 produced the highest correlation, 0.785, which was statistically significant. The other three products had correlations ranging from -0.058 to 0.291, and were not significantly different than zero. Thus, at least three products (*n* - 1) meet the no-correlation assumption and Proposition 3 holds. The closeness of the average consumption ratios with the weighted average consumption ratios also provides supportive evidence for the no-correlation assumption. Thus, ABC product costs can be determined without a first-stage allocation.

Summary and Conclusion

Simplifying ABC is a key objective if we expect ABC to be more widely adopted and/or sustained. We have shown that it is possible, through after-the-fact simplification, to create smaller, simpler, and less-costly systems with no loss of product costing accuracy relative to a complex ABC system. Therefore, it is possible to reduce the cost and complexity of ABC even without reducing its benefits. Using the properties of linear dependence and independence we show that parsimonious systems may exist where the product cost assignments are essentially equal to an assignment by an ABC system, where the number of cost drivers chosen for the simplified system is no greater than the number of products of the firm. Extending beyond this result, we also find that it is possible to reduce the number of cost drivers in a parsimonious cost system below the number of products. This is shown by using the properties of vector subspaces and the concept of spanning.

The results found in this study significantly extend previous research as the accuracy reduction of simplified systems is no longer a valid assumption. Indeed, systems do exist where aggregation of cost pools and drivers can occur without any loss in accuracy compared to an ABC system with a large amount of complexity.

TABLE 9 Sample for correlation evidence

Activity	Consumption Ratios ($\boldsymbol{\delta}_{ij}$) Product 1	Product 2	Product 3	Product 4	Activity Cost
1	0.089	0.182	0.286	0.444	\$393
2	0.109	0.191	0.336	0.363	\$628
3	0.118	0.233	0.379	0.270	\$909
4	0.084	0.195	0.381	0.340	\$269
5	0.110	0.171	0.397	0.321	\$871
6	0.118	0.155	0.289	0.437	\$808
7	0.117	0.188	0.326	0.369	\$781
8	0.085	0.210	0.396	0.309	\$584
9	0.112	0.210	0.329	0.349	\$455
10	0.090	0.190	0.346	0.374	\$347
Correlation[a]	0.785*	−0.058	0.127	0.291	
Sample $\overline{\boldsymbol{\delta}_i}$[b]	0.102	0.193	0.346	0.357	
Sample $\boldsymbol{\varepsilon}_i$[c]	0.107	0.192	0.348	0.352	

[a]Calculated using activity cost and the consumption ratios of each product.
[b]The average of the consumption ratios of each product column.
[c]The activity-based product cost divided by the total cost of all ten activities: $\frac{\sum_{j=1}^{10} \delta_{ij} c_j}{\$6{,}045}$.
*Significantly different from zero, with $p < 0.01$.

This study has addressed after-the-fact cost system simplification. Future research possibilities include a focus on before-the-fact simplification to address how implementation costs can be reduced. Future research in this area may determine if sampling or correlation analysis will reduce the steps and, therefore, the cost of implementation. Sampling techniques may be useful for decreasing the size of the original ABC system. A random sample of activities can produce very good estimates of the complete set of consumption ration, where the sample size is significantly smaller than the total number of activities. Another possibility is eliminating or reducing first stage allocations by investigating the nature of the correlation between consumption ratios and activity costs. If there is no correlation between consumption ratios and activity costs, it may be possible to determine the cost assignment without performing a first stage allocation. These techniques may help future companies embrace the adoption of ABC.

Notes

1. Aggregation error results from reducing the number of activity cost pools by adding costs of heterogeneous activities to form fewer and more aggregate cost pools. Specification error results from using the wrong cost driver to assign costs. Measurement error refers to incorrect assignment of costs to individual activities. Error is measured relative to an "unobservable" completely specified ABC system.
2. Throughout the remainder of the article, vectors are indicated by bold type.
3. If $\sum_{j=1}^{4} k_j \delta_j = 0$ only for $k_j = 0$ for all j, then the consumption ratio vectors are linearly independent.
4. It is possible that even if individual expected consumption ratios change, $\boldsymbol{\varepsilon}$ may remain unchanged. Thus, k is a minimum value for $\boldsymbol{\varepsilon}$-stability.

References

Abernathy, M., Lillis, A. M., Brownell, P., & Carter, P. (2001). Product Diversity and Costing System Design Choice: Field Study Evidence. *Management Accounting Research*, 12(3), 261–279.

Anderson, S. W., & Young, M. (1999). The Impact of Contextual and Process Factors on the Evaluation of Activity-Based Costing Systems. *Accounting, Organizations, and Society*, 24(7), 525–559.

Babad, Y. M., & Balachandran, B. V. (1993). Cost Driver Optimization in Activity-Based Costing. *The Accounting Review*, 68(3), 563–575.

Cagwin, D., & Bouwman, M. J. (2002). The Association between Activity-Based Costing and Improvement in Financial Performance. *Management Accounting Research*, 13(1), 1–39.

Cokins, G. (2002). Activity Based Costing: Optional or Required?, *AACE International Transactions*, RI31–RI36.

Cooper, R. & Kaplan, R. S. (1988). Measure Cost Right: Make the Right Decision. *Harvard Business Review*, 66, 96–103.

Datar, S., & Gupta, M. (1994). Aggregation, Specification and Measurement Errors in Product Costing. *The Accounting Review*, 69(4), 567–591.

Gosselin, M. (1997). The effect of strategy and organizational structure on the adoption and implementation of activity-based costing, *Accounting Organizations and Society*, 22, 105–122.

Homburg, C. (2001). A Note on Optimal Cost Driver Selection in ABC. *Management Accounting Research*, 12(2), 197–205.

Innes, J., Mitchel, F., & Sinclair, D. (2000). Activity-Based Costing in the U.K.'s Largest Companies: A Comparison of 1994 and 1999 Survey Results. *Management Accounting Research*, 11(3), 349–362.

Itner, C. D., Lanen, W., & Larcker, D. F. (2002). The Association between Activity-Based Costing and Manufacturing Performance. *Journal of Accounting Research*, 40(3), 711–726.

Kennedy, T., & Affleck-Graves, J. (2001). The Impact of Activity-Based Costing Techniques on Firm Performance. *Journal of Management Accounting Research*, 13(1), 19–46.

Krumwiede, K. R. (1998). ABC: Why It's Tried and How It Succeeds. *Management Accounting*, 79(10), 32–38.

Labro, E., & Vanhoucke, M. (2008). Diversity in Resource Consumption Patterns and Robustness of Costing Systems to Errors. *Management Science*, 54(10), 1715–1730.

Pryor, T. (2013). Simplify Your ABC. *Integrated Cost Management Systems, Inc. Newsletter*. Retrieved from http://icms.net/simplify-your-abc/

About the Authors

Dr. Shannon Charles is an Associate Professor, Lecturer at the University of Utah. She received her Ph.D. in accounting from Oklahoma State University.

Dr. Don Hansen is a Professor (retired) at Oklahoma State University. He received his Ph.D. in accounting from the University of Arizona.