MIS Quarterly

RESEARCH NOTE

ASSESSING COMMON METHOD BIAS: PROBLEMS WITH THE ULMC TECHNIQUE[1]

Wynne W. Chin
C. T. Bauer College of Business, University of Houston, Houston, TX 77204, U.S.A., and Department of Service Systems Management and Engineering, Sogang University, Seoul, KOREA {wchin@uh.edu}

Jason Bennett Thatcher
College of Business and Behavioral Science, Clemson University, Clemson, SC 29634-0701 U.S.A. {jthatch@clemson.edu}

Ryan T. Wright
Isenberg School of Management, University of Massachusetts, Amherst, Amherst, MA 01003 CA 94117 U.S.A. {rtwright@isenberg.umass.edu}

Recent work, in journals such as MIS Quarterly *and* Management Science, *has highlighted the importance of evaluating the influence of common method bias (CMB) on the results of statistical analysis. In this research note, we assess the utility of the unmeasured latent method construct (ULMC) approach in partial least squares (PLS), introduced by Liang et al. (2007). Such an assessment of the ULMC approach is important, because it has been employed in 76 studies since it appeared in MIS Quarterly in early 2007. Using data generated via Monte Carlo simulations, we use PLS structural equation modeling (SEM) to demonstrate that the ULMC approach of Liang et al. is neither able to detect, nor control for, common method bias. Method estimates using this approach resulted in negligible estimates, regardless of whether there were some, large, or no method bias introduced in the simulated data. Our study contributes to the IS and research methods literature by illustrating that, and explaining why the ULMC approach does not accurately detect common method bias in PLS. Further, our results build on prior work done using covariance-based SEM questioning the usefulness of the ULMC technique for detecting CMB.*

Keywords: Common method bias, unmeasured latent method construct, partial least squares, structural equation modeling

[1]Chris Higgins was the accepting senior editor for this paper. Andrew Burton-Jones served as the associate editor.

The authors contributed equally and therefore were listed alphabetically.

The appendices for this paper are located in the "Online Supplements" section of the *MIS Quarterly*'s website (http://www.misq.org).

Introduction

Papers often offer both theoretical and methodological contributions to the Information Systems (IS) literature. This includes the introduction of new frameworks, which contributes to theory. For example, Davis' (1989) introduction of the technology acceptance model also drew attention to the theory of reasoned action as a framework. Papers contribute to methods when they offer guidelines for, or serve as exemplars of, how to conduct analysis. Davis's work provided an example of how to validate new construct measures (e.g., perceived usefulness and perceived ease of use). By making contributions to theory and methods, papers reshape how we theorize about information systems while also demonstrating how to assess the strength of the theory's ability to describe reality.

Recently, *MIS Quarterly* published a paper by Liang et al. (2007) that offered a contribution to our understanding of enterprise systems. In terms of theory, the paper explained how institutional forces interplay with top management to shape post-implementation assimilation of enterprise systems by organizations. Due to its theoretical implications, the findings of Liang et al. have been cited in conference proceedings such as the International Conference on Information Systems, articles in journals including *MIS Quarterly, Information Systems Research,* and *Management Science*, and in several dissertations (see Appendix A).

What is notable about the work of Liang et al., not unlike Davis's early work, is that the paper's reach has extended to shape methods employed to detect common method bias (CMB). Drawing on Podsakoff et al. (2003) and Williams et al. (1989), Liang et al. introduced an *ad hoc* unmeasured latent marker construct (ULMC) approach that uses partial least squares (PLS) structural equation modeling (SEM) to detect and control for the influence of common method bias on analysis. CMB is systematic variance attributable to common measurement artifacts that alter (e.g., inflate or deflate) correlations in the underlying constructs (see Burton-Jones 2009; Liang et al. 2007; Malhotra et al. 2006; Murphy et al. 2004; Podsakoff et al. 2003; Sharma et al. 2009). ULMC is one method in the family of techniques used to control and/or detect CMB (Lindell and Whitney 2001; Richardson et al. 2009; Spector 2006; Williams et al. 2010). By enabling researchers to parse out trait and method error, Liang et al. claimed to offer a ULMC approach that detects whether CMB influences the results of analysis using PLS. See Appendix A for a list of commonly used CMB detection techniques.

Perhaps because of the simplicity and intuitive appeal of the ULMC approaches of Liang et al., it has rapidly diffused across many IS literature streams beyond enterprise systems. The ULMC approach has been cited in at least 76 papers since its publication (see Appendix B). Across these studies, few authors have detected CMB's presence in their data, and, if they did, they argue that the ULMC method suggests its influence is negligible (see Vance et al. 2008). In these 76 papers, we found no evidence that the ULMC detected moderate or high levels of CMB. That the ULMC approach of Liang et al. has been so rapidly embraced by authors suggests that it has the potential to become part of the "normal" toolkit used by IS researchers to detect the influence of CMB. While we applaud Liang et al. for attempting to articulate a solution to CMB, the authors did not did conclusively prove their ULMC procedure works either in a closed form numerical proof or even via a reasonable set of simulation runs.

Using Monte Carlo simulation to generate known true score data that conforms to models with or without CMB, we demonstrate that the ULMC approach does not correctly estimate, nor does it compensate for, the effect of CMB in PLS. Our simulations, which varied levels of CMB and reliability of measures, suggest that researchers who employ the ULMC approach may conclude that CMB does not influence results when, in fact, it does. Our overall conclusion is that, regardless of whether CMB with differing impact levels exists or whether the reliabilities of measures are heterogeneous or equivalent in measuring the underlying construct, this ULMC approach ends up with the same conclusion: there is no CMB. Our findings imply that researchers who use the ULMC approach in PLS may come away with the incorrect belief that there is minimal or no CMB, and lead the literature astray by making inappropriate inferences from their results.

Hence, this research note constitutes a correction to the record. It begins by describing the ULMC approach of Liang et al. and explaining why it is necessary to evaluate this method using PLS SEM. Next, we explain how we constructed our Monte Carlo simulations and evaluate 10 different scenarios that applied ULMC in PLS. Then, we report the results and describe problems tied to this quickly diffusing ULMC method that appeared in *MIS Quarterly*. We conclude by discussing why PLS is less efficacious at detecting or controlling CMB than covariance-based SEM using the ULMC approach, even though the ULMC technique is quite problematic when using covariance-based SEM too (Richardson et al. 2009).

The ULMC Approach

The ULMC approach draws upon the MTMM idea of modeling an underlying CMB construct. Instead of creating a separate set of indicators that reflect the CMB, the ULMC is modeled by specifying factor loadings from the method factor to any or all other items in the model suspected of being contaminated by CMB. This model, which includes a CMB factor, is then estimated using covariance-based SEM (Richardson et al. 2009, p. 769). ULMC and other latent marker approaches have been widely diffused in behavioral and business research (Lindell and Whitney 2001; Podsakoff et al. 2003; Richardson et al. 2009; Williams et al. 2003; Williams et al. 2010).

Liang et al. introduced an ULMC approach to test and control for CMB using PLS. According to Liang et al., four steps are necessary to implement a ULMC approach in PLS.[2] First, this approach requires taking all of the indicators for each construct and reusing them to create single indicator constructs. Second, the researcher links the original constructs to their respective single indicator constructs. Then, the method construct consisting of all indicators used in the study is linked to all the single indicator constructs (see Figure 1). Finally, researchers estimate a model using bootstrapping.

To apply ULMC in PLS, Liang et al. suggest examining the "coefficients of [each indicator's] two incoming paths from its substantive construct and the method factor" (p. 87). They maintain that "these two path coefficients are equivalent to the observed indicator's loadings on its substantive construct and the method factor and can be used to assess the presence of common method bias" (p. 87). Citing Williams et al. (2003), Liang et al. argue that

> evidence of common method bias can be obtained by examining the statistical significance of factor loadings of the method factor and comparing the variances of each observed indicator explained by its substantive construct and the method factor. The squared values of the method factor loadings were interpreted as the percent of indicator variance caused by method, whereas the squared loadings of substantive constructs were interpreted as the percent of indicator variance caused by substantive constructs. If the method factor loadings are insignificant and the indicators' substantive variances are substantially greater than their method variances, we can conclude that common method bias is unlikely to be a serious concern (p. 87).

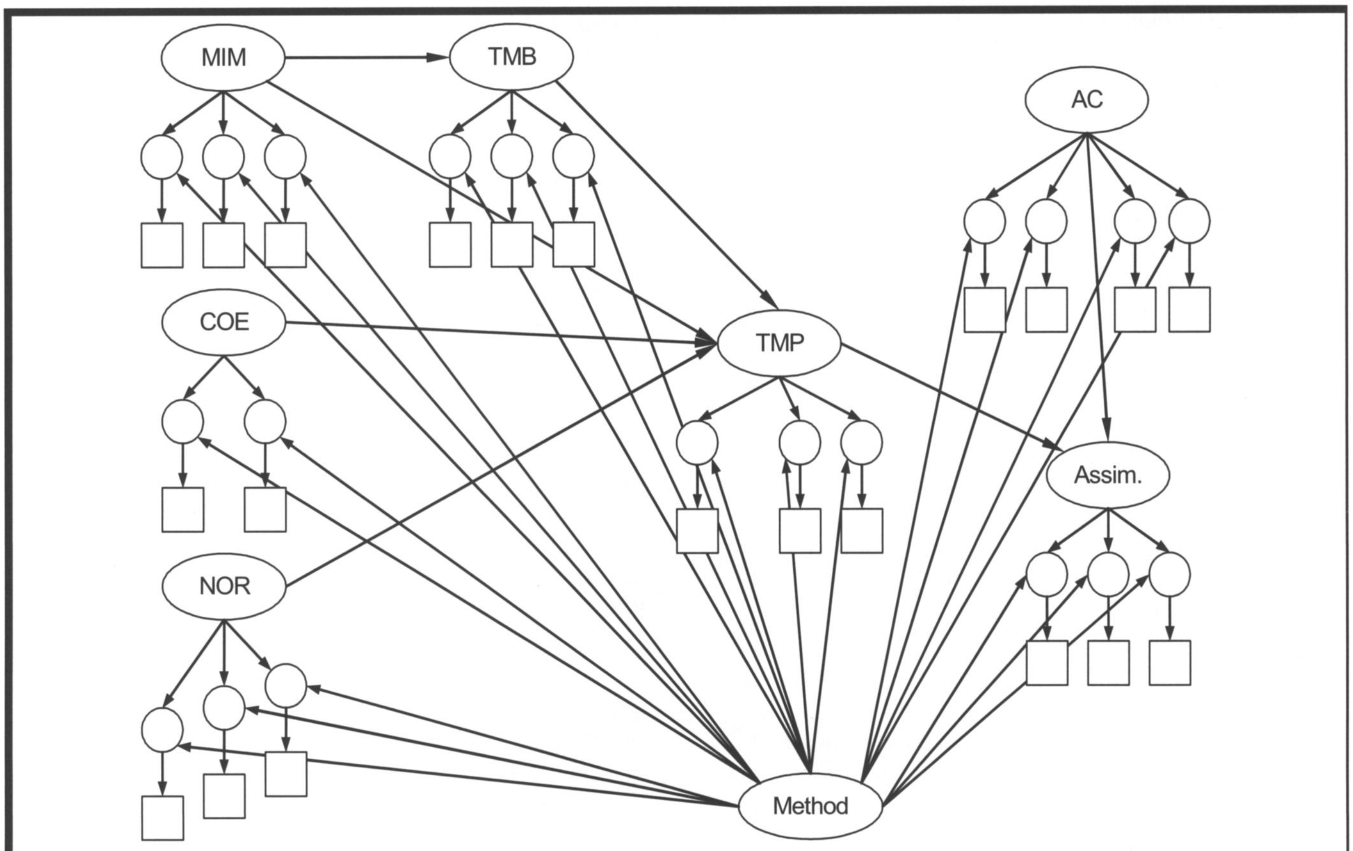
[2]Although Liang et al. do not provide explicit detail on how they implemented their ULMC approach, Vance et al. (2008) explicate on their multistep process in PLS. For more detail on this analysis, please see Liang et al. or Vance et al.'s online appendices.

Although Liang et al. do not present rigorous statistical vetting of their new ULMC method for detecting CMB, this approach has rapidly diffused and appears in respected IS research outlets. It has been used in studies that are published in top IS journals such as *MIS Quarterly*, *Information Systems Research*, *Journal of the Association for Information Systems*, and *Management Science*. It is also finding its way into other top journals such as the *Journal of Marketing*. Moreover, this method seems to have been quickly embraced by junior scholars in master's theses and dissertations (see Appendix B). This is understandable, because junior scholars are often the first to adopt innovative analytic techniques as a means to demonstrate the timeliness and rigor of their work.

Rigorously evaluating the PLS ULMC approach is important for several reasons. First, while the covariance-based SEM and PLS ULMC approaches are represented in a graphically similar manner, they differ in how they estimate models. Where covariance-based SEM evaluates model covariance fit, the PLS estimation technique maximizes variance explained in complex measurement and structural models. Consequently, when covariance-based SEM may struggle to estimate covariance matrices implied by complex ULMC models (Podsakoff et al. 2003), PLS, due to its ability to estimate "packages of variables and aggregate parameters," may more readily handle the "large, complex models with latent variables" (Wold 1985, p. 589) necessary for using ULMC to detect CMB.

Second, these differences in model estimation influence how one uses ULMC to test, or control, for CMB. A covariance-based SEM ULMC approach compares the fit between the implied covariances from the model estimates and the sample data covariances (Kline 2005). To detect CMB's influence, a covariance-based ULMC approach relies on comparisons of model fit. In contrast, using regression, PLS simultaneously estimates a system of linear relationships to maximize variance in relationships from indicators and theoretical constructs as well as relationships between constructs (Chin 1998b). To detect CMB's influence, a PLS ULMC approach relies on tests of loadings and their significance. Perhaps this is why Liang et al. reasoned that PLS offers an alternative method for partialing out CMB, as the algorithm calculates the weight relations among the variables rather than using maximum likelihood to estimate the fit of the data to the theoretical model; however, they offered no evidence that PLS can actually partial out CMB.

Third, PLS ULMC and covariance-based ULMC are operationalized in different ways. In covariance-based SEM, a method factor is operationalized as a first-order construct, which includes a path to every item in the model. However,



MIM = Mimetic Forces; COE = Coercive Forces; NOR = Normative Forces; TMP = Top Management Participation; AC = Absorptive Capacity; Assim. = Assimilation

Figure 1. Liang et al. Model for Assessing ULMC (Figure E3 from Liang et al., 2007, p. 86)

because a method factor may present identification problems, attempts to detect CMB in covariance-based SEM do not always work (Podsakoff et al. 2003). When using PLS, a method factor is operationalized as a PLS-based second-order construct (Chin 2010, p. 666). Liang et al. cite two reasons for these "finesses" to the ULMC approach. First, PLS does not accommodate random error. Because the PLS algorithm maximizes the explained variance, all error is accounted for by the estimation technique. Second, practically speaking, they argue that PLS Graph 3.0 and similar software packages do not accommodate an item to be determined by more than one construct.[3] As a result, the PLS ULMC approach requires that "all major constructs of interest and the method factor become second-order constructs" (p. 85; see Figure 2). Because a second-order PLS ULMC approach is substantially different from a first-order covariance-based ULMC approach, it is important to evaluate its ability to detect or control for CMB under different conditions.

Fourth, it is unclear whether PLS can detect, or control for, congeneric and noncongeneric CMB. According to Richardson et al., researchers make one of two assumptions regarding CMB's distribution in a sample. First, the noncongeneric perspective assumes that CMB has equal effects on all constructs within the nomological network. Second, the congeneric perspective assumes that CMB's effects vary based on the "nature of the rater, item, construct, and/or context. As such, one or more method constructs will be differentially correlated with substantive items and constructs" (Richardson et al. 2009, p. 766). Due to factor interdeterminancy problems (e.g., a complex model may not be identified or con-

[3]Strictly speaking, this second rationale is not true. Similar to principal components analyses, any items utilized in PLS Graph 3.0 can be estimated as being influenced by one or more orthogonal PLS components. If a researcher specifies a second or higher dimensional analysis, the variance of each item would then be modeled as being impacted by the PLS components from each dimension (Lohmöller 1989).

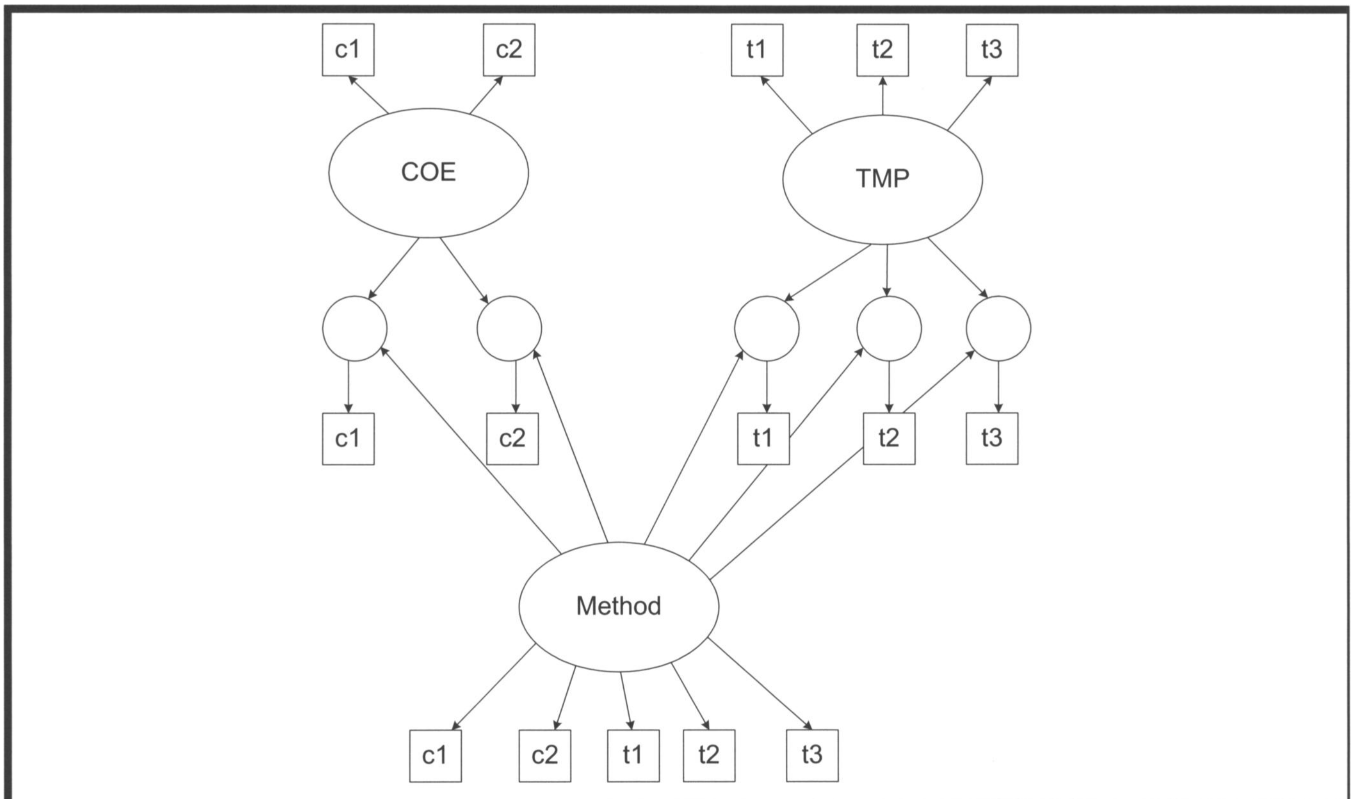


Figure 2. Example of ULMC Approach

verge), the type of contamination has important implications when applying covariance-based SEM detection techniques. For this reason, the CMB will have to be modeled differently depending on assumptions about its form (e.g., congeneric or noncongeneric) (Podsakoff et al. 2003; Richardson et al. 2009). Because PLS estimates latent variables as exact linear combinations of the observed measures (Chin et al. 2003), it, conversely, holds the potential for detecting or controlling for CMB's influence on estimates and/or constructs (regardless of the CMB's form) without changing the modeling assumptions. Although PLS has the potential to estimate congeneric and noncongeneric models, there is a lack of evidence that a PLS ULMC approach can detect different forms of CMB.

Fifth, the utility of the ULMC approach has been questioned in the broader research methods literature. After using a simulation to compare CMB detection techniques, Richardson et al. observed that "when used in data with CMV, the ULMC technique was almost always the least accurate at detecting CMV and bias" (p. 794). Their ULMC covariance-based analysis correctly identified CMB 41 percent of the time. This led Richardson et al. to conclude that "it is highly risky to use the ULMC approach for detection and to improve the accuracy of conclusions drawn about hypothesized relationships" (p. 794). They call for their results to be replicated and extended using other methods including regression-based approaches (p. 797). Given Richardson et al.'s admonition that the ULMC approach be used with caution in covariance-based SEM, it is understandable that we have seen a steady increase of this technique, utilizing PLS software, in top IS journals since 2009.

Finally, within the IS literature, research has only begun to investigate the utility of the ULMC approach. Bagozzi (2011) suggests that it is "unclear whether significant loadings on the method factor actually represent or correct for method bias" (p. 278). Moreover, Bagozzi suggests that the ULMC approach could introduce unintended consequences such as overfitting the model, improper parameter estimates, or out-of-range factor loadings in covariance-based SEM. It is important to note that Bagozzi does not mention PLS or Liang et al.'s ULMC approach.

Hence, the decision of Liang et al. to develop a PLS-based ULMC approach was reasonable, because of the properties of the estimation technique. However, because the utility of

ULMC approaches has been questioned in covariance-based SEM (Richardson et al. 2009, p. 794) and the PLS-based method loading approach is unvetted (Bagozzi 2011), we conducted a series of simulations to assess whether ULMC in PLS actually detects, or controls for, CMB.

Evaluating the ULMC Approach Using Monte Carlo Simulation

The use of Monte Carlo simulation affords an opportunity for assessing the effectiveness of the ULMC approach in PLS. In normal research settings, the investigator knows neither the true model relationships nor the true individual sample-case level scores for each construct in the model. The researcher can only make estimates using the item measures from the study's empirical data. In contrast, Monte Carlo simulation begins with *a priori* "known" (i.e., population) parameters of the model relationships and uses that information to generate data. The item-level data mirroring empirically gathered data would include varying levels of statistical noise and possibly other effects (e.g., CMB). These data are then used to see how well a proposed algorithm is able to estimate (i.e., recover) the true population parameters. Thus, through the use of Monte Carlo-simulated data, a researcher knows exactly what the true model relationship effects are and can test how well a statistical procedure is able to estimate it.

In the following steps, we describe our application of Monte Carlo simulation to evaluate the PLS UMLC approach. First, we explain our choice of simulation techniques. Next, we describe the heuristics that guided how we used Monte Carlo to create the data. Then, we provide detail on 10 different scenarios that varied the conditions (e.g., factor loadings and common method bias) used to assess the ULMC approach. Finally, we present the results of our simulation.

Selecting a Monte Carlo Technique

There are two common approaches that SEM researchers have used when generating Monte Carlo data (Reinartz et al. 2002). In the first approach, the implied covariance matrix of the observed variables is computed on the parameter the researcher specifies for the model. Then data are generated on the observed variables from a multivariate distribution having this covariance matrix. For the second approach, data are first generated to represent the true latent variables' scores according to the relationships specified in the model. Subsequently, data are generated on the observed variables from the latent variables in the model (i.e., item indicators). The second approach, in particular, is better suited for testing the PLS algorithm since the goal of PLS is to generate estimates of the true construct scores using the item-level data and the model relationships (e.g., paths) are estimated using these estimated scores. Using such data, we can then compare how close these estimates are to using the actual "true" scores. Hence, we used the second technique in our simulation.

Monte Carlo Parameters

True-value scores included in the Monte Carlo simulation adhered to best practices suggested in the SEM literature about the number of indicators, loadings, and error. By doing so, we conducted a conservative test of ULMC's ability to detect or control for CMB in PLS under ideal conditions. This approach included utilizing the best possible research design for detecting CMB utilizing PLS. At the item level, the data were created using the following heuristics:

1. When creating data, our constructs were represented by six indicators. Our use of six indicators per construct reflects a conservative interpretation of guidelines for estimating structural equation models found in the literature. Scholars such as Bollen (1989) suggest that the effective use of SEM requires using a minimum of three manifest indicators per construct. Reflecting a more cautious approach, Chin et al. (2003) suggested using six manifest indicators when evaluating a new method in PLS. Therefore, even though we were conducting a simulation, we adopted the more conservative approach to modeling our constructs in order to provide a rigorous test of the ULMC method in PLS.

2. We adhered to prescriptions in the literature of simulation and methods when setting item loadings (Chin et al. 2003; Goodhue et al. 2007; Hair et al. 2006). For the noncongeneric data, our indicators' loadings were set to .70. For the congeneric data, our indicator's loadings were set to an average .70 (with two set at .80, two set at .70, and two set at .60). We did so for three reasons. First, loadings greater than .60 are thought to be of practical significance (Hair et al. 2006). Second, to be used in SEM analysis, methodologists suggest item loading should be greater than .70 (Bollen 1989; Hair et al. 2006). When loadings are .70 or greater, the latent construct explains more of the variance in the item than the error. Third, a .70 loading is consistent with simulations that have evaluated the efficacy of PLS applications in prior IS research (Goodhue et al. 2007).

3. We ensured that our error was randomly distributed along a normal distribution within samples. As noted by Siemson et al. (2010), "if CMV inflates a correlation, it will at the same time deflate the standard error of this correlation. If CMV deflates a correlation, it will at the same time inflate the standard error of this correlation" (p. 471). By using randomly generated error, we emulated CMB's influence on data collected in a natural setting.

4. To model CMB, we selected three levels of method effects. Our first level was zero or no CMB. We did this for two reasons: first, so that we could check how closely the simulated data corresponds to the true population parameter; second, so we could illustrate what a path would look like in a PLS model that lacked a method factor. Our second level was .16. This level was incorporated into this study because Malhotra et al. (2006) suggest that the most likely method effect in IS research has a variance around .16 (e.g., .40 method loading). The third level for CMB variance was .36. We selected this level because it represents a high level of common method bias (e.g., .60 method loading) (Richardson et al. 2009; Williams et al. 2003).

At the model level, the data reflected the following assumptions:

1. Our model included only the relationship between two constructs. This approach is consistent with past simulations of covariance-based ULMC (see Richardson et al. 2009) as well as past PLS techniques (see Chin et al. 2003; Goodhue et al. 2007). Moreover, drawing on our conservative approach, this simple model allows for the best chance to detect CMB over more complex models.

2, We set a population parameter estimate of .60. By doing so, we created a scenario where we could emulate different levels of CMB in the data while retaining our ability to estimate the true population parameter. Given that common method bias may account for between 16 and 27 percent of variance observed (Cote and Buckley 1987; Doty and Glick 1998; Sharma et al. 2009; Williams and Brown 1994), a lower parameter estimate would have limited our ability to manipulate its level in our data and conduct meaningful tests of the UMLC approach's efficacy. In addition, estimating a .60 population parameter allowed us to evaluate CMB's influence against a backdrop of a large effect size (Cohen 1988), which is a circumstance under which reviewers often complain that CMB influences results (Pace 2010).

Creating the Data

We generated a series of simulated data using Monte Carlo techniques that provide different levels of method variance. Following procedures consistent with past studies (see Chin et al. 2003; Mattson 1997), we used PRELIS 2.14 (Jöreskog and Sorbom 1993) to generate true constructs scores, true CMB scores, and random noise in order to form item measurement data that conform to different underlying population models (i.e., varying degrees of CMB and reliability of measures; see Appendix C for details on how we generated our data). For example, an item that is designed to reflect an underlying true construct with 0.70 standardized loading, 0.40 standardized loading from method bias, and the rest from noise would be modeled as

$$\text{Item} = 0.70 \times \text{LV} + 0.4 \times \text{CMB} + 0.591608 \times \text{NRAND}$$

where LV, CMB, and NRAND represent the underlying latent construct, common method effect, and random noise, respectively. All error terms are set such that the overall variance sums to 1.0 for each indicator.

For each scenario, we generated 500 datasets of 5,000 cases. With the control allowed by generating simulated data, we saved the actual latent scores. This allows us to check how closely the simulated data at 5,000 cases corresponds to the true population path, which represents the asymptotic level (i.e., the population or infinite sample size situation) set at 0.60 using PRELIS. Thus, when evaluating the ULMC approach, we can test the actual true path relationship given our large sample size as well as how close we get to this number when we no longer have the actual latent scores and must use the item measures. By using large sample sizes to evaluate the ULMC approach per scenario, we minimize the chance that statistical fluctuation confounds our results for each PLS run and maximize statistical power across our analyses (Chin et al. 2003).

Monte Carlo Scenarios

The scenarios varied in the following combinations of characteristics: CMB simulated, latent measure type, and CMB measure type (see Table 1). As noted above, CMB was at three levels—0 or none, .16, and .36—across the scenarios. The latent measure type for the construct items was either congeneric or noncongeneric. The CMB measure type was either congeneric, noncongeneric, or method score (M score). Where the congeneric and noncongeneric CMB measures utilize all of the indicators to represent the underlying method factor, the Method (M) score represents the true CMB gener-

Table 1. Summary of the Scenarios for PLS ULMC Analysis

Scenarios	Amount of CMB Simulated	Latent Measure Type	CMB Measure Type	True Latent Construct Loading	Common Method Loading
1	0	Noncongeneric[a]	Noncongeneric	.70	0
2	.16	Noncongeneric	Noncongeneric	.70	.40
3	.36	Noncongeneric	Noncongeneric	.70	.60
4	.16	Congeneric[b]	Noncongeneric	.80, .70, .60	.40
5	.36	Congeneric	Noncongeneric	.80, .70, .60	.60
6	.36	Noncongeneric	Congeneric	.70	.20, .40, .60
7	.36	Congeneric	Congeneric	.80, .70, 0.6	.20, .40, .60
8	0	Noncongeneric	M Score	.70	0
9	.16	Noncongeneric	M Score	.70	.40
10	.36	Noncongeneric	M Score	.70	.60

[a]Noncongeneric assumes that the loadings are equal for all items on a construct.
[b]Congeneric assumes that loadings vary based on the rater, item, construct, and/or context.

ated in our simulation. Where one does not know the true CMB in the field data, we know the exact value of the M score in the simulated data and can represent it in one item, rather than reusing several items. By using the actual M score, we can test how well the ULMC-proposed partitioning of latent trait, method effect, and random error works when using the "exact" measure of the "method" factor that was employed to create the indicators in the first place. In other words, by using the actual M score, we eliminate questions about whether the aggregating of all the indicators in the congeneric or noncongeneric conditions represent good approximations for CMB (see Little et al. 2002). Through manipulating these three characteristics in the simulated data, we are able to examine the ULMC's ability to detect and partial out CMB using PLS.

PLS Analysis and Results

In the following analysis, we will demonstrate that the PLS ULMC approach seems to result in negligible method estimates regardless of the amount of CMB. In other words, whether a large amount, a moderate amount, or no systematic method bias exists, our analysis suggests the same conclusion of no method effect is obtained. This is also true whether we kept the impact of the CMB constant for all measures (i.e., noncongeneric measures) or if it the impacts varied (i.e., congeneric).

Using PLS-Graph 3.0, we illustrate these problems with the ULMC approach by estimating (1) a baseline with no common method effect, (2) scenarios with method effects in the item loadings, and (3) scenarios with single-item common method effects (e.g., M score). Our analyses are presented in the following order: three noncongeneric scenarios, four congeneric and noncongeneric mixed scenarios, and noncongeneric and M score mixed scenarios. The actual relationships between variable X with variable Y are the true scores generated in the population; this is the true finite sample (albeit quite large) to the population baseline model of 0.60. Further, for the structural model where latent variable XX influences latent variable YY, the average of 500 runs (i.e., sample sets) per scenario are given. In the remainder of this article, we utilize X and Y to denote the true score variables, whereas XX and YY represent the estimated latent variables. Further, the loadings on the XX and YY constructs will be defined as the trait loadings, whereas the loadings from the method construct will be defined as the method loadings.

Figure 3 gives an example model utilized in our scenarios. In this figure, you can see the relationship between the XX and its underlying constructs, which are related to the single-item measure, respectively. For example, construct A1, which is derived from the single-item measure item 1 (i1), is regressed on both the method effect and the XX construct. Path weights from XX to the A1 construct are considered the factor loadings, whereas path weights from the method to the A1 construct are considered the method effect.

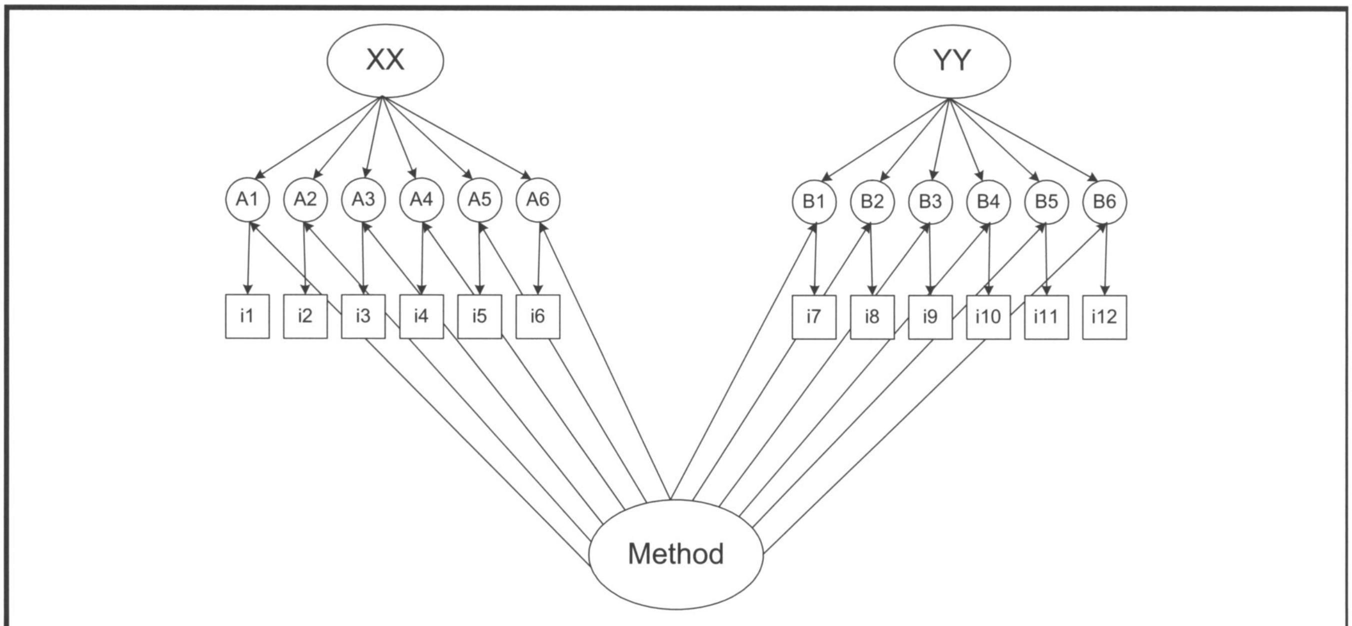


Figure 3. Example Structural Model

Tables 2 and 3 present the results from all 10 scenarios. Specifically, we first show the mean and standard deviation from each type of simulation for the path weight from the true X construct to the true Y construct. Next, we report the path weight between XX and YY when including the method effects in the model. Further, these tables convey each path weight from both the method effect and the substantive latent construct. The findings for each scenario are outlined below.

Scenario 1: No Common Method Effect, Noncongeneric Measures

Our first analysis represents a baseline situation. Here, both constructs have six indicators and are set with identical loadings of 0.70 (i.e., noncongeneric measures), implying that the underlying constructs (whether latent variable or common method) have the same impact on all measures. The linkage between the latent scores is very close to the population value of 0.60. If one were conducting analysis using non-simulated data, this would be the path estimated without a method factor in the model. If we were to increase the sample size by tenfold, we suspect the estimate would be literally at 0.60. But, using the suggested procedure, the estimated path is 0.495 and the resulting path estimates suggest that the method effect consisting of all the indicators has very little impact, as one would expect if no method source existed. Notice that the trait loadings are higher than the population setting of 0.7 by approximately 10 percent. This is due to a known bias in the PLS algorithm, as highlighted by Chin in his workshop lectures and articles, which increases the loadings (Chin 1998a, 1998b; Chin and Gopal 1995; Chin et al. 2003). This reflects what is known statistically as *consistency at large* where PLS tends to underestimate the correlation between the latent variables and overestimate the loadings. This bias disappears only when the number of indicators per latent variable and the number of cases become very large (Lohmöller 1989). Now, consider if these trait loadings were higher than 10 percent. This might be an indication of additional effects beyond the accuracy of the PLS algorithm such as the ULMC method. Further, as per the recommendation of Liang et al., we evaluate the path weights from the method to the specific constructs (e.g., Method ➔ A1) to see if there is a significant common method effect for this item. The path weight of .015 is not statistically substantive. Using Liang et al.'s heuristic, this would suggest CMB has not contaminated the relationship between the construct and the item.

Scenario 2: Common Method Effect (.16 Variance), Noncongeneric Measures

Scenario 2 introduces a low level of common method effect of around 0.16 (Malhotra et al. 2006). As before, each indicator had an identical loading of 0.70. However, each indicator now included a 0.40 loading to a common method factor. The

Table 2. Summary Results of the Scenarios' PLS ULMC Analysis (Scenarios 1 through 5)*

Scenario	S1		S2		S3		S4		S5	
Path	Mean	Std. Dev.	Mean	Std. Dev.	Mean	Std. Dev.	Mean	Std. Dev.	Mean	Std. Dev.
X →Y (true score)	0.596	0.009	0.596	0.009	0.599	0.009	0.601	0.009	0.595	0.009
XX→YY	0.495	0.011	0.636	0.008	0.748	0.006	0.640	0.008	0.752	0.006
XX→A1	0.758	0.019	0.796	0.017	0.927	0.013	0.869	0.014	0.965	0.007
XX→A2	0.741	0.019	0.884	0.017	0.909	0.013	0.874	0.013	0.965	0.007
XX→A3	0.773	0.019	0.834	0.018	0.952	0.013	0.844	0.017	0.949	0.014
XX→A4	0.732	0.017	0.834	0.018	0.951	0.013	0.828	0.018	0.921	0.013
XX→A5	0.769	0.019	0.853	0.016	0.941	0.010	0.813	0.020	0.927	0.018
XX→A6	0.779	0.019	0.850	0.017	0.933	0.013	0.822	0.021	0.884	0.019
YY→B1	0.734	0.019	0.842	0.017	0.908	0.013	0.876	0.014	0.956	0.006
YY→B2	0.756	0.018	0.842	0.016	0.935	0.014	0.889	0.013	0.956	0.006
YY→B3	0.758	0.017	0.849	0.018	0.938	0.013	0.823	0.017	0.919	0.014
YY→B4	0.746	0.019	0.867	0.018	0.928	0.014	0.846	0.018	0.943	0.013
YY→B5	0.752	0.018	0.802	0.016	0.951	0.013	0.806	0.022	0.897	0.018
YY→B6	0.782	0.018	0.850	0.017	0.946	0.014	0.799	0.020	0.952	0.018
M (Method)→A1	0.015	0.021	-0.040	0.019	0.025	0.014	0.030	0.014	0.021	0.007
M→A2	-0.006	0.021	0.053	0.018	0.010	0.014	0.034	0.015	0.021	0.007
M→A3	-0.006	0.020	0.000	0.020	-0.016	0.012	0.000	0.018	-0.015	0.015
M→A4	0.034	0.018	0.010	0.019	-0.015	0.014	0.014	0.019	0.018	0.014
M→A5	-0.013	0.021	-0.011	0.018	-0.008	0.010	-0.039	0.021	-0.047	0.019
M→A6	-0.023	0.021	-0.012	0.019	0.004	0.014	-0.051	0.023	-0.003	0.019
M→B1	0.011	0.021	0.000	0.017	0.030	0.014	0.026	0.015	0.032	0.007
M→B2	-0.003	0.019	0.001	0.018	-0.002	0.014	0.011	0.014	0.032	0.007
M→B3	0.002	0.017	-0.007	0.019	-0.004	0.014	0.018	0.019	0.018	0.014
M→B4	0.019	0.021	-0.028	0.019	0.007	0.015	-0.004	0.019	-0.003	0.014
M→B5	0.002	0.020	0.043	0.018	-0.018	0.014	-0.038	0.023	-0.014	0.018
M→B6	-0.030	0.020	-0.009	0.018	-0.013	0.015	-0.021	0.021	-0.072	0.019

***Scenario 1 (S1)** = Latent Item Loadings (LIL) are noncongeneric (NC), Method Loadings (ML) are 0,
S2 = LIL are NC, ML are NC at .4,
S3 = LIL are NC and ML are NC at .6,
S4 = LIL are congeneric (C) and ML are NC at .4,
S5 = LIL are C and ML are NC at .6.

Table 3b. Summary Results of the Scenarios' PLS ULMC Analysis (Scenarios 6 through 10)*

Scenario	S6		S7		S8		S9		S10	
Path	Mean	Std. Dev.	Mean	Std. Dev.	Mean	Std. Dev.	Mean	Std. Dev.	Mean	Std. Dev.
X →Y (true score)	0.587	0.009	0.596	0.009	0.596	0.009	0.596	0.009	0.599	0.009
XX→YY	0.635	0.008	0.646	0.008	0.495	0.011	0.636	0.008	0.748	0.006
XX→A1	0.905	0.020	0.953	0.017	0.752	0.007	0.847	0.006	0.936	0.005
XX→A2	0.926	0.021	0.924	0.018	0.753	0.006	0.850	0.005	0.930	0.005
XX→A3	0.896	0.017	0.856	0.017	0.768	0.006	0.832	0.006	0.940	0.004
XX→A4	0.880	0.015	0.866	0.017	0.762	0.006	0.843	0.006	0.945	0.005
XX→A5	0.751	0.013	0.723	0.018	0.757	0.006	0.841	0.006	0.930	0.005
XX→A6	0.734	0.014	0.710	0.017	0.759	0.007	0.838	0.006	0.933	0.004
YY→B1	0.878	0.019	0.935	0.018	0.743	0.007	0.843	0.006	0.933	0.004
YY→B2	0.936	0.021	0.933	0.018	0.754	0.006	0.841	0.006	0.929	0.005
YY→B3	0.866	0.017	0.863	0.017	0.760	0.006	0.843	0.006	0.935	0.005
YY→B4	0.869	0.017	0.833	0.017	0.762	0.006	0.847	0.006	0.931	0.005
YY→B5	0.746	0.014	0.729	0.018	0.754	0.006	0.833	0.006	0.944	0.005
YY→B6	0.762	0.013	0.736	0.017	0.756	0.007	0.844	0.006	0.934	0.005
Method (M)→A1	-0.184	0.022	-0.117	0.019	0.001	0.010	-0.005	0.009	0.005	0.007
M→A2	-0.155	0.022	-0.144	0.019	0.001	0.009	-0.007	0.009	0.001	0.007
M→A3	-0.058	0.019	-0.010	0.018	0.008	0.009	0.006	0.009	-0.004	0.006
M→A4	-0.032	0.017	-0.024	0.018	0.000	0.010	0.000	0.009	-0.014	0.007
M→A5	0.173	0.014	0.135	0.019	-0.012	0.009	0.004	0.009	0.005	0.007
M→A6	0.191	0.015	0.149	0.018	0.003	0.009	0.003	0.009	0.006	0.006
M→B1	-0.132	0.021	-0.118	0.019	0.013	0.010	-0.002	0.009	0.005	0.006
M→B2	-0.203	0.022	-0.129	0.019	0.007	0.009	0.003	0.008	0.007	0.007
M→B3	-0.027	0.018	-0.020	0.019	-0.008	0.009	-0.002	0.009	-0.001	0.007
M→B4	-0.033	0.018	0.010	0.018	0.000	0.009	-0.012	0.009	0.005	0.007
M→B5	0.173	0.015	0.125	0.018	-0.010	0.010	0.019	0.009	-0.015	0.007
M→B6	0.158	0.014	0.123	0.018	-0.001	0.010	-0.006	0.009	-0.001	0.007

***Scenario 6 (S1)** = Latent Item Loadings (LIL) are noncongeneric (NC), Method Loadings (ML) are congeneric (C) at an ave. of .4,
S7 = LIL are C, ML are C at an average of .4,
S8 = LIL are NC and ML are represented by the method (M) score at 0,
S9 = LIL are NC and ML are represented by the method (M) score at .4,
S10 = LIL are NC and ML are represented by the method (M) score at .6.

0.40 loading squared represents the 0.16 variance contributed by the common method. Throughout the remaining scenarios, random noise was added (e.g., error) to each indicator to create standardized items.

The path between the latent scores for this set of simulated data, once again, shows close approximation to the population path of 0.60. Tables 2 and 3 show the results of using the ULMC approach. Interestingly, the common method paths should be closer to 0.40 if the ULMC technique works correctly. Instead, the estimated method effect indicates negligible CMB. But notice that the loadings are higher. On average, the loadings in this model are 0.85, whereas the previous loadings under conditions of no method effect were around 0.76. The difference in variance would be 0.85 × .085 – .076 × .076 = 0.145, which is close to the 0.16 variance placed into the indicators. Thus, by not accurately estimating the method effect, the trait loadings are inflated by using the ULMC technique. In turn, the structural path using these measures with common method bias is now higher at 0.636 than the true population setting of 0.60. Further, per Liang et al.'s recommendation, we evaluated the loadings from the method to the specific constructs (e.g., Method ➔ A1) to see if there is a significant common method effect for this item. The path loading of .015 is not substantive, which, according to the heuristic of Liang et al., suggests that CMB has not contaminated this relationship.

Scenario 3: Common Method Effect (.36 Variance), Noncongeneric Measures

In our final noncongeneric scenario, we evaluated a model with a higher level of CMB. We estimated the same baseline model, but increased the common method loading to 0.60 (e.g., .36 CMB). The close approximation of the population path of 0.60 suggests that our simulation of true scores was set-up properly (see Figure 4). The common method paths should be around 0.60, but, as represented in Figure 5, these paths were once again negligible. The construct loadings are even higher—on average, at the 0.94 level. The difference in variance relative to no method bias would be 0.94 × .94 – .076 × .076 = 0.30, which approximates the 0.36 variance we placed onto each indicator. The structural path with these measures is also correspondingly higher than the true path at 0.748. Thus, our simulations suggest that the ULMC approach is ineffective. The paths from the method construct are estimated as trivial regardless of whether the amount of method bias was 0, 0.16, or 0.36. This is striking, because we know that if the ULMC method worked, it should detect CMB in the inflated item loadings. This pattern of non-significant method loadings recurs throughout our simulations.

Scenario 4: Common Method Effect (.16 Variance), Congeneric Measures

In Scenario 4, we simulated congeneric indicator loadings with the method effect set to .16 (similar to Scenario 2 with noncongeneric measures). Instead of being fixed at 0.70, the first two indicators were set to 0.80, the next two remained at 0.70, and the last two were lowered to 0.60. Thus, while the average loading remains at 0.70, the reliabilities of each indicator are no longer equivalent. The path between the latent scores for this set of simulated data of 0.601, once again, shows a close approximation to the population path of 0.60. The results are very similar to the Scenario 2 results where the model parameters are almost identical except for the congeneric loading from the measures to their underlying constructs (see Table 2). Again, the common method paths should be closer to 0.40, if the ULMC technique works correctly. Instead, the estimated method effects indicate it is negligible.

Scenario 5: Common Method Effect (.36 Variance), Congeneric Measures

Scenario 5 retained the congeneric loadings of Step 4, but increased the method effect to 0.36 (similar to Scenario 3 with noncongeneric measures). The path between the latent scores for this set of simulated data of 0.594 shows a close approximation to the population path of 0.60. The common method paths should be closer to 0.60, if the ULMC technique were able to detect or control for it. Instead, the path between the two latent constructs was inflated from a true 0.60 parameter to an estimate of 0.752. Hence, the results of using the ULMC approach with congeneric measures are very similar to those obtained in Scenario 3.

Scenario 6: Common Method Effect (Congeneric Loadings), Noncongeneric Measures

For Scenario 6, we returned to noncongeneric construct loadings of 0.70, but varied the method effect. Instead of fixing the common method at 0.4 or 0.6, the first two indicators' common method effects were set at 0.2, the next two at 0.4, and the last two at 0.60. Thus, while the average CMB loading remains at 0.40, the effects on each measure are no longer equivalent. The path between the latent scores for this set of simulated data of 0.586, once again, shows a close approximation to the population path of 0.60. The results of using the ULMC approach are very similar to those obtained in Scenarios 2 and 4 where the method variance was set at 0.16 with an inflated path estimate of 0.635. If the technique

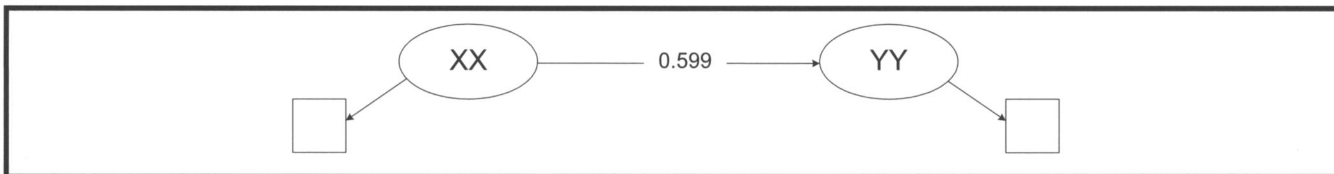
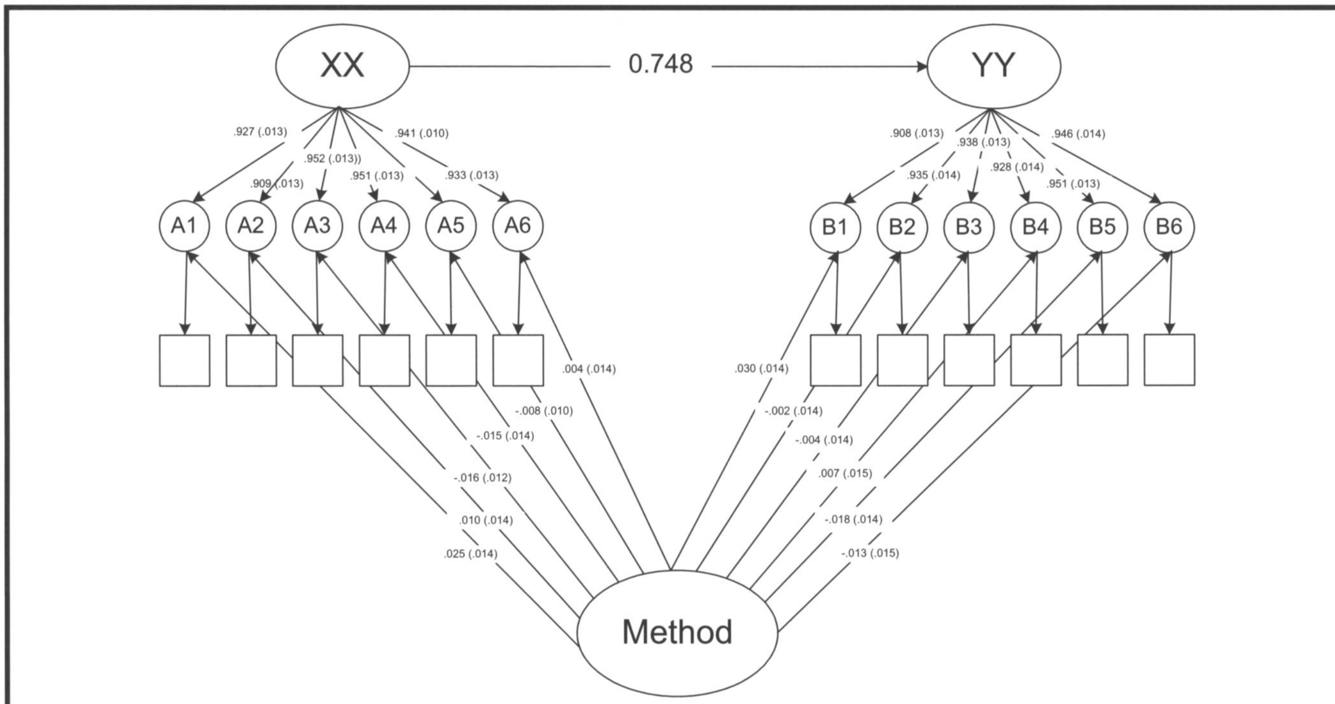


Figure 4. Scenario 3: True Score Model



*Path weights are the average of 500 samples in the scenario.
**Standard deviations are included in parentheses.

Figure 5. Scenario 3: Structural Model with .36 CMB Variance

accurately estimated the loadings from the method factor, we should see 0.2, 0.4, or 0.6 from each set of items. Instead, the estimated loadings never exceeded an absolute magnitude of 0.2 and incorrectly suggest the first four items for each construct would be negative (see Table 3). In turn, the loading estimates to the underlying constructs were inflated. Even when the common method varies, the PLS ULMC approach does not accurately detect and control for the CMB effects.

Scenario 7: Common Method Effect (Congeneric Loadings), Congeneric Measures

For our final scenario using congeneric data, we varied both the loadings for the underlying construct and the method effect. The first two items for each construct had true score and method loadings of 0.8 and 0.2, respectively. This was followed by 0.7 and 0.4 for the next two items. The final two were set at 0.6 and 0.6 (i.e., equal amounts of true and method effects). The path between the latent scores for this set of simulated data of 0.596, once again, shows a close approximation to the population path of 0.60. The results of using the ULMC approach again resulted in an inflated path estimate of 0.647. The estimated method effect never exceeded an absolute magnitude of 0.15 and incorrectly suggests the first four items for construct XX and the first three items for construct YY would be negatively impacted by the method effect. Once again, the ULMC approach is unable to accurately detect or control for the CMB effects.

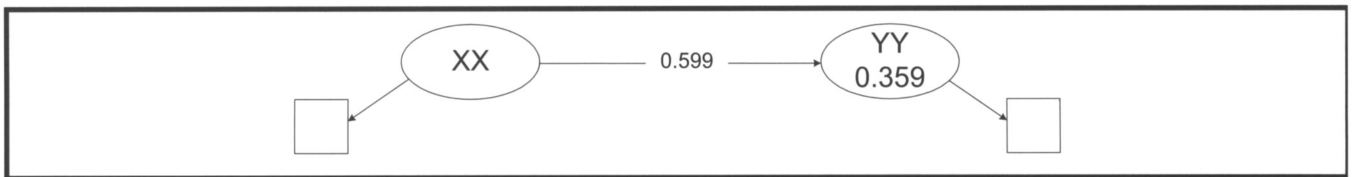
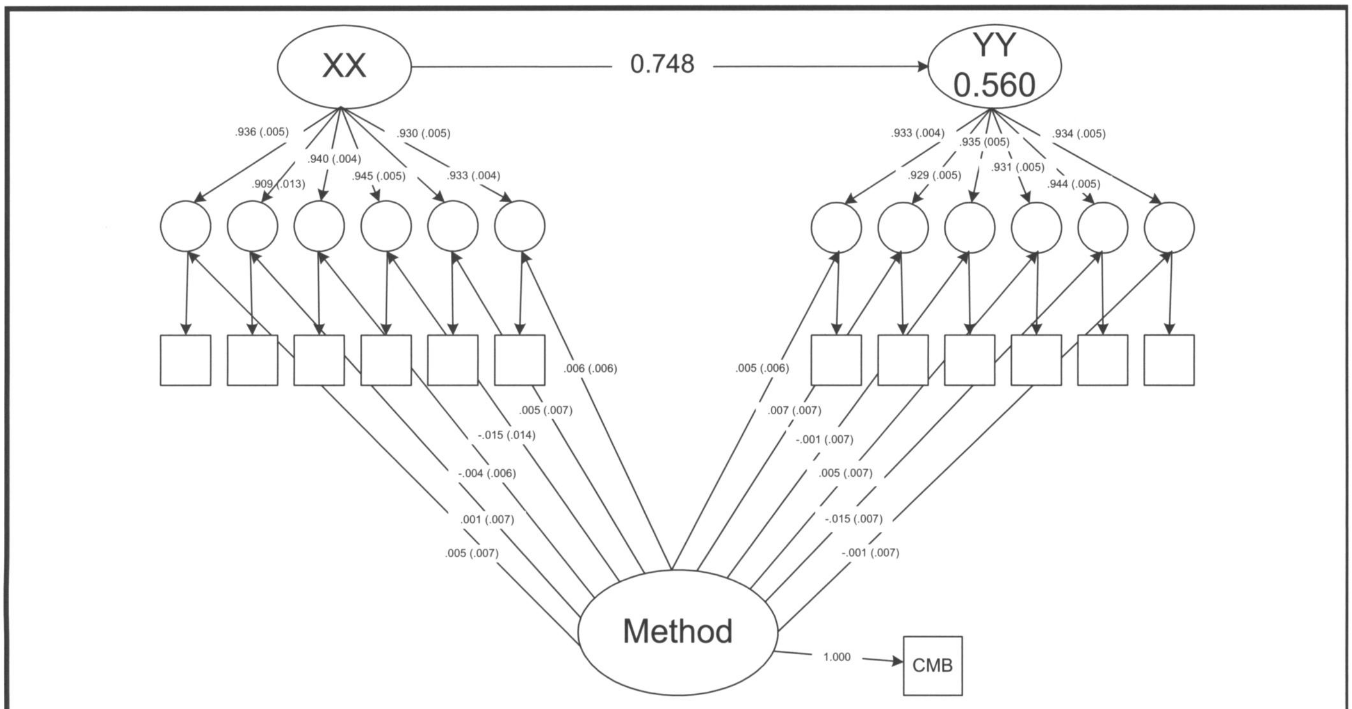


Figure 5. Scenario 10: True Score Model



*Path weights are the average of 500 samples in the scenario.
**Standard deviations are included in parentheses.

Figure 5. Scenario 10: True Score Model with CMB at .36 Variance

Scenarios 8, 9, and 10: Assessing the ULMC Technique Using M Scores

To further probe the utility of the ULMC technique, we remove one of the constraints of the procedure: accurate estimation of the common method. To do so, the next three scenarios (8, 9, and 10) replace the common method loadings with the true M score. Thus, with the actual method effect provided, we can test how well the ULMC technique performs in assessing the impact of the method on each item and on partialing out the method effect from the XX and YY trait constructs with the goal of reducing potential path inflation. The path between the latent scores for this set of simulated data, once again, shows a close approximation to the population path of 0.60 (see Figure 6 for Scenario 10). The results of using the ULMC approach again resulted in an inflated path estimate. Figure 7 illustrates how the M score was incorporated in the model when .36 CMB variance was added to the method path loadings. We also kept the loadings to the underlying construct fixed to 0.70 to allow us to compare with results from Scenarios 2 and 3. By fixing the construct loadings to .70 and using the M score, we remove the chance that random error confounds the results of our Monte Carlo simulation scenarios. Because we eliminated all additional sources of unexplained variance, this represents a conservative test of ULMC's ability to detect CMB. We found that even when using the actual M score at moderate (.16) or high (.36) levels of CMB, ULMC does not detect or control for CMB's influence on the true parameter estimates (see Table 3).

In sum, our conclusion is that the ULMC approach seems incorrect. In very large samples, it does not accurately estimate the method effect when using the aggregated items. Nor does it work even if we have the true M scores. Our analysis leads to the conclusion that the ULMC approach using PLS does not detect or control for CMB and that the structural paths and the indicators' loadings are biased upward as the method bias increases.

Discussion of the ULMC Technique

Where prior covariance-based SEM simulations have demonstrated that the ULMC procedure does not consistently identify CMB, our PLS SEM simulations suggest that the Liang et al. instantiation of ULMC does not accurately detect, or control, for CMB. Through analysis of 10 scenarios that varied the reliability of the construct loadings and type of CMB under ideal conditions, we provide evidence that the ULMC technique of aggregating existing measures does not detect CMB in PLS. This finding contrasts with Richardson et al., who reported that covariance-based SEM ULMC approaches correctly identify CMB, "about 41% of the time." Because such low accuracy rates " rarely meet the criteria for usefulness," Richardson et al. recommend that readers do not use ULMC approaches to detect CMB (p. 794). Given that we found even more severe problems with ULMC in PLS, our findings suggest that the IS community abandon the ULMC technique of Liang et al. and seek alternative means to detect or control for CMB.

So the question remains, is the ULMC procedure problematic or is the operationalization in PLS of Liang et al. problematic? We argue that while the ULMC technique when operationalized in covariance-based SEM has difficulties detecting and controlling for CMB (Richardson et al. 2009), the same technique applied in PLS has no ability to detect and control CMB. We offer three reasons for the failure of a PLS-based ULMC approach.

First, while the PLS model is represented as graphically similar to the CB SEM, we believe the PLS procedure Liang et al. describe is inherently problematic. The primary reason is the difference in the underlying PLS algorithm as applied to the same graphical model heretofore only applied using the CB SEM algorithm. Specifically, PLS uses a component-based procedure where every construct in the model eventually is estimated as a weighted average of its indicators, while the CB SEM algorithm estimates all parameters in the model with the objective of getting the implied covariances to closely match the sample covariances. Therefore, even though Figure 1 may give the impression that a method construct is partialing out the effects of CMB from each indicator, this is not true in PLS. Each of the traits examined by the Liang et al. model (e.g., mimetic forces [MIM], coercive forces [COE], and normative forces [NOR]) are modeled to load on the reused single indicator constructs while also being composed of the same set of indicators. For example, COE has two reflective indicators (see Figure 2). However, due to the finesse introduced by Liang et al., COE is also modeled with two paths to two single indicator constructs. Therefore, those indicators used to form the COE trait construct still contain CMB. Thus, any structural paths among any of the trait constructs, including COE, will still be biased by any method effect that exists.

Second, although using PLS allows one to circumvent issues tied to model indeterminacy and complexity, it does not address other potential problems with the estimate of the method effect. Liang et al.'s suggestion of using all indicators in the study to estimate CMB influence in a model is problematic. This approach, as modeled, in fact represents a multidimensional construct comprising all the traits' indicators along with any potential CMB. Clearly, for more complex models with more traits, the estimated method construct will primarily represent traits. Considering that CMB may be derived from many different sources (e.g., rater effects, item characteristic effects, and context effects) and vary at different levels (noncongeneric), it is not surprising that a single method factor, as demonstrated in our scenarios, may not effectively partial out CMB's influence in PLS.

Finally, the paths from the method construct to any single indicator construct will necessarily be minimal due to the component approach of PLS. Consider the COE construct as an example (see Figure 2). We know that PLS forms a COE component as a weighted combination of its two indicators. Then, the COE component score is used to predict each of the two reused single indicator constructs along with the method construct. We would expect a weighted sum of two indicators used to predict any one of the two indicators to be high. The method construct, in contrast, being a weighted sum of all indicators in the model, would be so diluted as to have minimal predictive impact. In fact, we would expect the more indicators used in a model, the more likely the negligible path estimates from the method construct, implying no CMB.

Conclusion

Addressing problems with the approach of Liang et al. for managing CMB is important, given that their ULMC technique has been used in 13 theses and dissertations, top journals (e.g., *MIS Quarterly*, *Information Systems Research*, and *Management Science*), as well as papers across the spectrum of issues examined by IS researchers. Considering that the PLS ULMC approach has been widely applied in IS research in just four years, we believe it is important for researchers to pause and rigorously assess its merits before it becomes a dominant method for detecting CMB. Through simulation, our study clearly calls into question the method's ability to detect or correct for the influence of CMB on results. Given the importance of minimizing CMB, there needs to be further research that evaluates both covariance-based and PLS-based techniques.

As a final point, we want to emphasize that the purpose of our research note is to prevent IS researchers being led astray by a problematic Appendix, not to challenge the core findings, of Liang et al.'s study. Lacking a reliable technique, we are unable to assess CMB's influence on the findings of Liang et al. As a result, we believe that researchers with domain-specific expertise should assess their paper's core theoretical contribution and, clearly, scholars studying ERP have found much merit in the study of Liang et al.

Acknowledgments

This research was partially supported by World Class University program funded by the Ministry of Education, Science and Technology through the National Research Foundation of Korea(R31-20002).

References

Bagozzi, R. P. 2011. "Measurement and Meaning in Information Systems and Organizational Research: Methodoloigcal and Philosophical Foundations," *MIS Quarterly* (35:2), pp. 261-292.

Bollen, K. A. 1989. *Structural Equations with Latent Variables*, New York: Wiley.

Burton-Jones, A. 2009. "Minimizing Method Bias Through Programmatic Research," *MIS Quarterly* (33:3), pp. 445-471.

Chin, W. W. 1998a. "Issues and Opinion on Structural Equation Modeling.," *MIS Quarterly* (22:1), p. vii.

Chin, W. W. 1998b. "The Partial Least Squares Approach to Structural Equation Modeling," in *Modern Methods for Business Research,* G. A. Marcoulides (ed.), Mahwah, NJ: Lawrence Erlbaum Associates, pp. 295-336.

Chin, W. W. 2010. "How to Write Up and Report PLS Analyses," in *Handbook of Partial Least Squares Concepts, Methods and Applications,* V. E. Vinzi, W. W. Chin, J. Henseler, and H. Wang (eds.), Berlin: Springer-Verlag, pp. 650-690.

Chin, W. W., and Gopal, A. 1995. "Adoption Intention in GSS: Relative Importance of Beliefs," *Data Base* (26:2/3), pp. 42-64.

Chin, W. W., Marcolin, B. L., and Newsted, P. R. 2003. "A Partial Least Squares Latent Variable Modeling Approach for Measuring Interaction Effects: Results from a Monte Carlo Simulation Study and an Electronic-Mail Emotion/Adoption Study," *Information Systems Research* (14:2), pp. 189-217.

Cohen, J. 1988. *Statistical Power Analysis for the Behavioral Sciences* (1st ed.), New York: Academic Press.

Cote, J. A., and Buckley, M. R. 1987. "Estimating Trait, Method, and Error Variance: Generalizing Across 70 Construct Validation Studies," *Journal of Marketing Research* (25), pp. 315-318.

Davis, F. D. 1989. "Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology," *MIS Quarterly* (13:3), pp. 319-340.

Doty, D. H., and Glick, W. H. 1998. "Common Method Bias: Does Common Method Variance Really Bias Results?," *Organizational Research Methods* (1:4), pp. 374-406.

Goodhue, D., Lewis, W., and Thompson, R. L. 2007. "Statistical Power in Analyzing Interaction Effects: Questioning the Advantage of PLS with Product Indicators," *Information Systems Research* (18:2), pp. 211-227.

Hair, J. F., Black, W. C., Babin, B. J., Anderson, R. E., and Tatham, R. L. 2006. *Multivariate Data Analysis*, Upper Saddle River, NJ: Prentice Hall.

Jöreskog, K. G., and Sorbom, D. 1993. *PRELIS 2 User's Reference Guide*, Lincolnwood, IL: Scientific Software International.

Kline, T. 2005. *Psychological Testing: A Practical Approach to Design and Evaluation* (1st ed.), London: Sage Publications.

Liang, H., Saraf, N., Hu, Q., and Xue, Y. 2007. "Assimilation of Enterprise Systems: The Effect of Institutional Pressures and the Mediating Role of Top Management," *MIS Quarterly* (31:1), pp. 59 - 87.

Lindell, M., and Whitney, D. 2001. "Accounting for Common Method Variance in Cross-Sectional Research Designs," *Journal of Applied Psychology* (86:1), pp. 114-121.

Little, T. D., Cunningham, W. A., Shahar, G., and Widaman, K. F. 2002. "To Parcel or Not to Parcel: Exploring the Question, Weighing the Merits," *Structural Equation Modeling: A Multidisciplinary Journal of Applied Psychology* (9:2), pp. 151-173.

Lohmöller, J.-B. 1989. *Latent Variable Path Modeling with Partial Least Squares*, Heidelberg, Germany: Physica-Verlag.

Malhotra, N. K., Kim, S. S., and Patil, A. 2006. "Common Method Variance in IS Research: A Comparison of Alternative Approaches and a Reanalysis of Past Research," *Management Science* (52:12), pp. 1865-1883.

Mattson, S. 1997. "How to Generate Non-Normal Data for Simulation of Structural Equation Models," *Multivariate Behavioral Research* (32), pp. 355-373.

Murphy, K. R., Cleveland, J. N., Skattebo, A. L., and Kinney, T. B. 2004. "Raters Who Pursue Different Goals Give Different Ratings," *Journal of Applied Psychology* (89:1), pp. 158-164.

Pace, V. L. 2010. "Method Variances From the Perspectives of Reviewers: Poorly Understood Problem or Overemphasized Complaint?," *Organizational Research Methods* (13:3), pp. 421-434.

Podsakoff, P. M., MacKenzie, S. B., Lee, J.-Y., and Podsakoff, N. P. 2003. "Common Method Biases in Behavioral Research: A Critical Review of the Literature and Recommended Remedies," *Journal of Applied Psychology* (88), pp. 879-903.

Reinartz, W. J., Echambadi, R., and Chin, W. W. 2002. "Generating Non-Normal Data for Simulation of Structural Equation Models Using Mattson's Method," *Multivariate Behavioral Research* (37:2), pp. 227-244.

Richardson, H. A., Simmering, M. J., and Sturman, M. C. 2009. "A Tale of Three Prespectives: Examining Post Hoc Statisical Techniques for Detection and Correction of Common Method Variance," *Organizational Research Methods* (12:4), pp. 762-800.

Sharma, R., Yetton, P., and Crawford, J. 2009. "Estimating the Effect of Common Method Variance: The Method-Method Pair Technique with an Illustration from TAM Research," *MIS Quarterly* (33:3), pp. 491-512.

Siemsen, E., Roth, A., and Oliveira, P. 2010. "Common Method Bias in Regression Models with Linear, Quadratic, and Interaction Effects," *Organizational Research Methods* (13:3), pp. 456-476.

Spector, P. E. 2006. "Method Variance in Organizational Research: Truth or Urban Legend?," *Organizational Research Methods* (9), pp. 221-232.

Vance, A., Elie-Dit-Cosaque, C., and Straub, D. W. 2008. "Examining Trust in Information Technology Artifacts: The Effects of System Quality and Culture," *Journal of Management Information Systems* (24:4), pp. 73-100.

Williams, L. J., and Brown, B. K. 1994. "Method Variance in Organizational Behavior and Human Resources Research: Effects on Correlations, Path Coefficients, and Hypothesis Testing," *Organizational Behavior and Human Decision Processes* (57), pp. 185-209.

Williams, L. J., Cote, J. A., and Buckley, M. R. 1989. "Lack of Method Variance in Self-Reported and Perceptions at Work: Reality or Artifact?," *Journal of Applied Psychology* (74), pp. 323-331.

Williams, L. J., Edwards, J. R., and Vandenberg, R. J. 2003. "Recent Advances in Causal Modeling Methods for Organizational and Management Research," *Journal of Management* (29:6), pp. 903-936.

Williams, L. J., Hartman, N., and Cavazotte, F. 2010. "Method Variance and Marker Variables: A Review and Comprehensive CFA Marker Technique," *Organizational Research Methods* (13), pp. 477-514.

Wold, H. 1985. "Partial Least Squares," in *Encyclopedia of Statistical Sciences*, S. Kotz and N. L. Johnson (eds.), New York: Wiley, pp. 581-591.

About the Authors

Wynne Chin is the C. T. Bauer Professor of Decision and Information Sciences at the University of Houston and the World Class University Professor in the Department of Service Systems Management and Engineering at Sogang University. He received an AB, MS, MBA, and Ph.D degrees in Biophysics, Biomedical/ Chemical Engineering, Business, and Computers and Information from the University of California–Berkeley, Northwestern University, and the University of Michigan, respectively. He is the developer of PLS-Graph, the first graphical based software dating back to 1990 to perform partial least squares analysis and used by more than 5,000 researchers worldwide. Wynne's research includes sales force automation, IT adoption, outsourcing, acceptance, satisfaction, group cohesion and negotiation, and psychometric and casual modeling issues. Wynne is on the editorial boards of *Structural Equation Modeling*, *Journal of Information Technology*, and *IEEE Transaction of Management*, and previously was coeditor of *Data Base* and on the boards of *Journal of the AIS*, *Information Systems Research*, and *MIS Quarterly*. Wynne currently resides in Sugar Land, TX, with his wife Kelly, his two daughters, Christina (22 years old) and Angela (17 years old), and his dog Bios (13 dog years).

Jason B. Thatcher is an associate professor in the Department of Management at Clemson University. He holds B.A.s in History (*cum laude*) and Political Science (*cum laude*) from the University of Utah as well as an M.P.A. from the Askew School of Public Administration and Policy and a Ph.D. in Business Administration from the College of Business at Florida State University. His research examines the influence of individual beliefs and characteristics on the use of information technology. Jason also studies strategic and human resource management issues related to the application of technologies in organizations. His work has appeared in *MIS Quarterly*, *Communications of the ACM*, *Journal of Management Information Systems*, *IEEE Transactions on Engineering Management*, *Organizational Behavior and Human Decision Processes*, and *Journal of Applied Psychology*. He has served on the editorial boards of *Information Systems Research*, *Journal of the AIS*, and *IEEE Transactions on Engineering Management*. Jason lives in Greenville, SC, where he enjoys eating moonpies, grits, and barbecue.

Ryan T. Wright is an assistant professor at the University of Massachusetts, Amherst. He holds a Ph.D. from Washington State University in Management Information Systems and an MBA and Bachelor of Science in Business from the University of Montana. Ryan's research interests take a behavioral approach to understanding how current technologies can be used to enable secure and efficient e-business transactions. He is published in the *Journal of MIS*, *Communications of the AIS*, and other peer-reviewed publications. In addition to academic achievements, Ryan's professional experience includes tenure as CTO of a successful startup, time in management at Amoco Oil (now BP), consulting projects for the U.S. Department of Commerce, and expert testimony in IS privacy and security. Ryan was in the 2008 ICIS Doctoral Consortium.