Journal of Experimental Psychology:
Human Perception and Performance
2007, Vol. 33, No. 1, 246–249

0096-1523/07/$12.00 DOI: 10.1037/0096-1523.33.1.246

OBSERVATIONS

The Contribution of Covariation to Skill Improvement Is an Ambiguous Measure: Comment on Müller and Sternad (2004)

Jeroen B. J. Smeets and Stefan Louw
Vrije Universiteit

It has been proposed that it is possible to decompose changes in variability of human motor behavior into 3 independent components: covariation, task tolerance, and stochastic noise (H. Müller & D. Sternad, 2004). The authors simulate learning to throw accurately and show that for this task the proposed analysis does not give an unambiguous answer to the question of what the 3 components contribute to the simulated skill improvement. It is argued that this is caused by the fact that the component covariation depends on the choice of control variables. The authors conclude that it is not possible to distinguish between the 3 components of noise reduction without knowing the controlled variables.

Keywords: motor control, learning, variability, compensation

It has been proposed that it is possible to split changes in variability of human motor behavior into three independent components: covariation, task tolerance, and stochastic noise (Müller & Sternad, 2004). If feasible, such decomposition would be very useful for understanding motor control, because the three components refer to clearly different processes. Reducing variability in performance by changing stochastic noise means low-level noise reduction. Exploiting task tolerance (finding a movement strategy in which the same noise has the least effect on performance) and covariation (compensating the variability in one execution variable by that in another variable) are two clearly distinct higher level processes. In this article we use a simulation of learning to throw accurately to study whether we can reliably determine the contribution of the three components.

Mathematical analysis is essential for a proper understanding of motor control. To gain such an understanding, one needs to translate both concepts and experimental variables into mathematical entities. If one of the assumptions made for this translation is not valid, a mathematical analysis can easily lead to erroneous conclusions about motor learning. For instance, if subjects reduce their movement time exponentially during learning, one might (erroneously) assume that the increase in peak velocity will also be exponential. An analysis based on this assumption will lead to the conclusion that there are several learning processes with different time constants in a situation in which there is only one learning process (Smeets, 2000). One common step in the translation is a transformation of variables, for instance from raw sensor data of a motion analysis system to Cartesian coordinates. So that we can draw conclusions about motor learning, the result of the analysis should be independent of the choice of variables, or it should be clear which variables should be used. Before going into the decomposition method proposed by Müller and Sternad (2004), we discuss this issue with a simple example.

If we want to know whether a subject's behavior is the same in two conditions, we generally start with comparing the means, a measure that is easy to calculate and understand. Comparing means is invariant under linear transformations such as a translation or rotation of the reference frame; nonlinear transformations, however, might affect comparisons of means. Let us assume that the subject had the same mean behavior in both conditions and that the variability was normally distributed. If we transform the data for the two conditions in a nonlinear way (e.g., by taking the logarithm), the distributions will no longer be normal. The consequence is that the means of the transformed variable will be different for the two conditions if the original standard deviations were different. The lesson of this example (a lesson that can be found in any textbook on statistics) is that comparing means is useful only if they are from a normal distribution. If this is not the case, one should either transform the data to get a normal distribution or use nonparametric statistics.

The use of the concept of covariation is not without pitfalls, as has been recognized by Müller and Sternad (2004). Therefore, instead of covariance they have used a "smart" method to calculate the contribution of covariation by permuting variables in execution space within a data set and studying the effect on the variability in task performance (Müller & Sternad, 2003). This method can deal with the effects of nonlinearity in the relation between variables. However, covariance has another problematic feature: It depends critically on the variables chosen to describe behavior. For instance, one can always rotate the frame of reference to get variables that have zero covariance. Is this also the case if we use the smart method for determining covariation? To see whether the choice of variables affects the extent to which changes in covariation (as determined by the method proposed by Müller & Sternad,

Jeroen B. J. Smeets and Stefan Louw, Faculteit Bewegingswetenschappen, Vrije Universiteit, Amsterdam, the Netherlands.

Correspondence concerning this article should be addressed to Jeroen B. J. Smeets, Faculteit Bewegingswetenschappen, Vrije Universiteit, Van der Boechorststraat 9, Amsterdam NL-1081 BT, the Netherlands. E-mail: j.smeets@fbw.vu.nl

2003) contribute to changes in task performance, we simulated learning to throw accurately.

For simplicity, we reduced the task space for throwing to only two execution variables: We restricted movements to a vertical plane and assumed that the release position was constant (vertically aligned at a distance [X] of 2 m from the target; see Figure 1). Thus only the two-dimensional release velocity varied between trials. The flight time in a trial (i) was equal to the horizontal distance divided by the horizontal component of the release velocity ($v_{x,i}$). The vertical distance moved depended on the flight time, vertical component of the release velocity ($v_{y,i}$), and the gravitational acceleration (g). The performance in each trial was equal to the absolute value of the vertical distance from the target; for a block of n trials, the performance measure D was the average error:

$$D = \sum_{i=1}^{n} \frac{1}{n} \left| v_{y,i} \frac{X}{v_{x,i}} - \frac{1}{2} g \left(\frac{X}{v_{x,i}} \right)^2 \right|.$$

We modeled three blocks (each 100 trials) and assumed that between these blocks subjects improved their performance by exploiting the task tolerance; we thus kept the (Gaussian) stochastic noise constant and the covariance zero (see Table 1 and Figure 1). The performance clearly improved (as shown by a lower value for D) by shifting the simulated movements to a region with a larger task tolerance ($\Delta D_{1,2} = 0.129$ m and $\Delta D_{2,3} = 0.091$ m). We simulated learning to throw using Cartesian coordinates, following the description of throwing in the original work (see Figure 1 of Müller & Sternad, 2004). The improvement in performance in our simulations was thus modeled as a pure exploitation of task tolerance, without any change in stochastic noise or covariance.

We subsequently applied the analysis proposed by Müller and Sternad (2004) using the polar coordinates they used to analyze their example task (skittles). As performance in both skittles and throwing is determined by release velocity, we found no reason not to analyze our throwing task also in polar coordinates. We transformed the simulated data from Cartesian coordinates (v_x, v_y) to the polar coordinates (ϕ, $|v|$; see Figure 2) and followed Müller and Sternad's method to decompose the changes in performance into changes in stochastic noise, covariation, and task tolerance. If the method yields results that are invariant under coordinate transformations, we should find the same results when analyzing in polar coordinates as when analyzing in Cartesian coordinates.

The method starts with determining the change in performance (ΔD) caused by covariation (ΔC) by permuting combinations of ϕ and $|v|$ within each of the three blocks of the experiment (Müller & Sternad, 2003). If covariation is exploited to obtain good performance, D will be larger for the permutated data. If D is smaller for the permutated data, the covariation hindered performance. In order to get a reliable measure for ΔC, we applied 10 different permutations and averaged the results. The difference in contribution of covariation to performance in two blocks is the contribution of covariation to the change in performance. Subsequently, the method determines the contributions of task tolerance (ΔT) by shifting the permutated data to the average location of the next block. If task tolerance is exploited, the same distribution of data will yield a better performance (lower value for D). The remaining change in performance is due to change in noise (ΔN):

$$\Delta D = \Delta C + \Delta T + \Delta N.$$

As expected for variables that are independent, we found negligible contributions of covariation and noise reduction in the

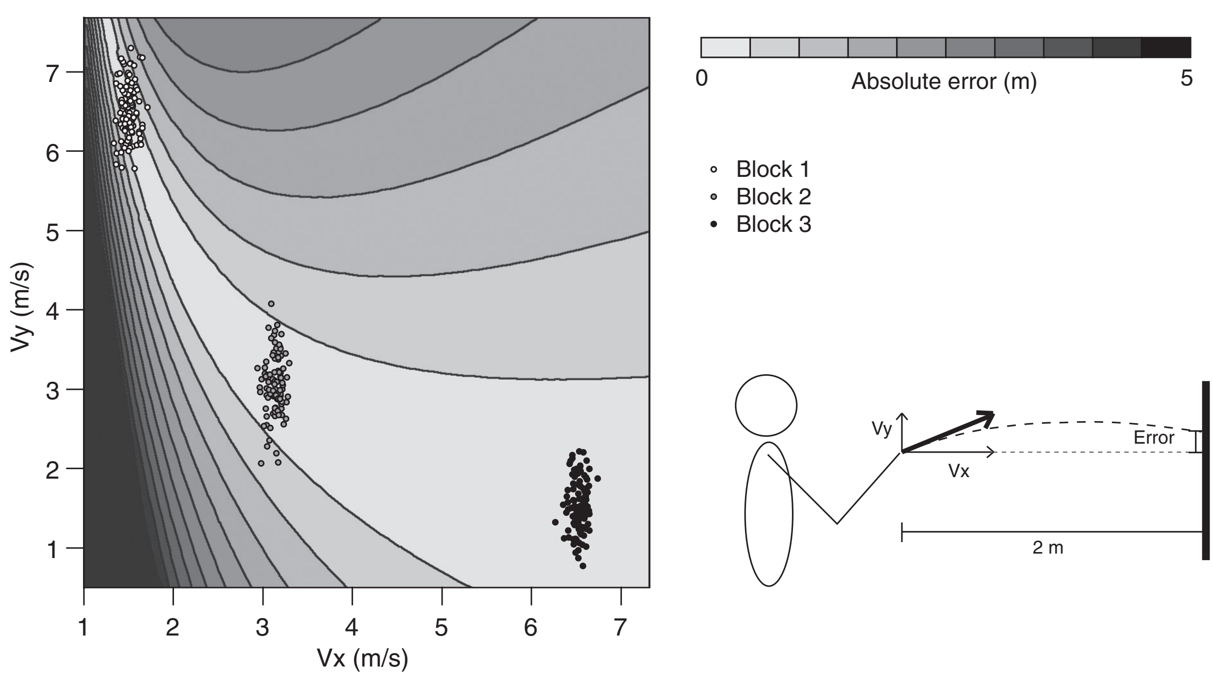

Figure 1. The graph (to the left) shows task space of throwing in Cartesian coordinates. Gray levels indicate the value of the absolute error on the target plane for each combination of release velocity (v). Each point represents a model trial. The human figure (to the right) illustrates the task geometry.

Table 1
Means and Standard Deviations of the Normal Distributions From Which the Release Parameters Were Randomly Chosen

Block	Horizontal release velocity (m/s)		Vertical release velocity (m/s)		
	M	*SD*	*M*	*SD*	*D* (m)
1	1.50 (1.49)	0.08 (0.07)	6.53 (6.53)	0.35 (0.34)	0.516
2	3.13 (3.12)	0.08 (0.08)	3.13 (3.06)	0.35 (0.37)	0.196
3	6.53 (6.52)	0.08 (0.07)	1.55 (1.55)	0.35 (0.33)	0.083

Note. The values for the resulting distributions appear in parentheses. D = resulting performance of the simulated throwing movements.

analysis based on Cartesian coordinates (see Table 2). The change in noise we found is due to the fact that the random choices of 100 samples from a Gaussian distribution had a slightly different standard deviation for each block (see Table 1). Analyzed in polar coordinates, however, the changes in performance reveal a considerable contribution of covariation and noise reduction. The change in covariation contributes negatively to improvement according to this analysis, which is compensated by noise reduction and a larger task tolerance. However, this is only so in polar coordinates. Described in Cartesian coordinates, our simulated learning was only due to exploiting task tolerance: The noise was almost constant and the contribution of covariation randomly fluctuated around zero.

Why does the apparent contribution of covariation depend on the coordinate system? It is a direct consequence of the fact that there is correlation between the two variables when using polar coordinates but not when using Cartesian coordinates. The transformation from Cartesian to polar coordinates is a nonlinear one, but even linear transformations can introduce covariance (and thus covariation). A simple example is a rotation of the Cartesian axes around the origin. Covariation in the data is therefore not a good measure of (lack of) compensation. One can always rotate the execution space to remove the covariance and thus change the covariation (this can be done for only one block at a time).

We have shown that a learning behavior that is described in Cartesian coordinates as a pure exploitation of the variations in task tolerance seems a lot more complex in polar coordinates when using the decomposition method for variability (Müller & Sternad, 2004). One property of a good mathematical tool is that its outcome should be reliable and not depend on the arbitrary choice of units or coordinate system. As the decomposition method yields results that clearly depend on the chosen coordinate system, it is not a very valuable tool. This is unfortunate, as the decomposition

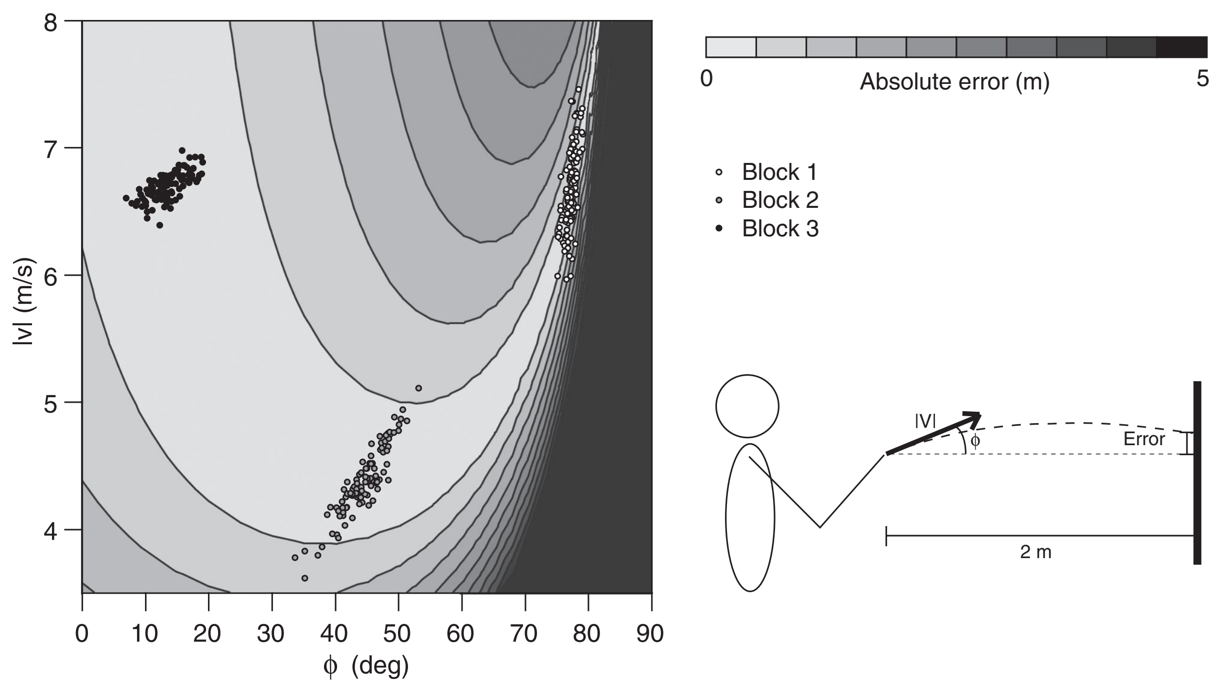

Figure 2. The graph (to the left) shows task space of throwing in polar coordinates. Gray levels indicate the value of the absolute error on the target plane for each combination of release velocity (v). Each point represents a model trial. The human figure (to the right) illustrates the task geometry.

Table 2
Average Change in Performance (D) and Components of Variability Between Trial Blocks

		Polar			Cartesian		
Block	ΔD	ΔC	ΔT	ΔN	ΔC	ΔT	ΔN
1→2	−0.320	0.326 ± 0.011	−0.598 ± 0.011	−0.049 ± 0.002	−0.004 ± 0.007	−0.326 ± 0.006	0.008 ± 0.002
2→3	−0.113	0.016 ± 0.001	−0.087 ± 0.002	−0.042 ± 0.001	0.001 ± 0.001	−0.108 ± 0.001	−0.006 ± 0.001

Note. Values are in meters (± SEM). Components of variability (covariation [*C*], task tolerance [*T*], and stochastic noise [*N*]) were determined with the method of Müller and Sternad (2004) in both polar and Cartesian coordinates, using 10 repetitions of the permutation.

method would be very useful if one could substantiate claims that one actually can tailor the (co)variance to the task.

Isn't it possible to save the method for decomposing variability? This might be possible if there is a "proper" coordinate system to apply the analysis for a given task. For comparing means, the proper coordinate system is the one in which the variability is distributed normally. For the decomposition of variability, a corresponding requirement for a proper coordinate system has not yet been defined. Probably, one can determine the proper coordinate system on the basis of the task. One might argue that this was the case in the experiment of Müller and Sternad (2004), because in their computer task subjects controlled two separate input devices. The controlled variables were speed profile of arm rotation (speed and position) and time of finger release. The authors did not use these variables, presumably because they form a three-dimensional space. In most daily tasks, it is not quite clear what the controlled variables are, neither at the level of the muscle (Stein, 1982) nor at the level of kinematics (Desmurget, Prablanc, Jordan, & Jeannerod, 1999; van den Dobbelsteen, Brenner, & Smeets, 2001). Even the proponents of the method use both a polar and a Cartesian coordinate system to describe throwing (Müller & Sternad, 2004). The only way to save the method is by leaving out the estimation of the covariation and decomposing the changes in performance in only two components: exploiting task tolerance and noise reduction. However, this decomposition does not work in all situations: Nonlinear transformations can also make the contribution of noise reduction ambiguous.

We have to conclude, therefore, that there is not yet a reliable way to decompose variability. Moreover, claims about covariation (or compensation) between variables do not describe a property of the behavior that one is studying but only a property of the variables that one has chosen to describe the behavior.

References

Desmurget, M., Prablanc, C., Jordan, M., & Jeannerod, M. (1999). Are reaching movements planned to be straight and invariant in the extrinsic space? Kinematic comparison between compliant and unconstrained motions. *Quarterly Journal of Experimental Psychology: Human Experimental Psychology, 52*(A), 981–1020.

Müller, H., & Sternad, D. (2003). A randomization method for the calculation of covariation in multiple nonlinear relations: Illustrated with the example of goal-directed movements. *Biological Cybernetics, 89,* 22–33.

Müller, H., & Sternad, D. (2004). Decomposition of variability in the execution of goal-oriented tasks: Three components of skill improvement. *Journal of Experimental Psychology: Human Perception and Performance, 30,* 212–233.

Smeets, J. B. J. (2000). The relation between movement parameters and motor learning. *Experimental Brain Research, 132,* 550–552.

Stein, R. B. (1982). What muscle variable(s) does the nervous system control in limb movements? *Behavioral and Brain Sciences, 5,* 535–577.

van den Dobbelsteen, J. J., Brenner, E., & Smeets, J. B. J. (2001). Endpoints of arm movements to visual targets. *Experimental Brain Research, 138,* 279–287.

Received November 4, 2005
Revision received April 30, 2006
Accepted May 2, 2006 ■