Behavior Research Methods (2024) 56:258–277
https://doi.org/10.3758/s13428-022-02042-9

A state response measurement model for problem-solving process data

Yue Xiao[1,2] · **Hongyun Liu**[2,3]

Accepted: 30 November 2022 / Published online: 3 January 2023

Abstract

In computer simulation-based interactive tasks, different people make different response processes to the same tasks, resulting in various action sequences. These sequences contain rich information, not only about respondents, but also about tasks. In this study, we propose a state response (SR) measurement model with a Bayesian approach for analyzing the process sequences, which assumes that each action made is determined by the individual's problem-solving ability and the easiness of the current problem state. This model is closer to reality compared with the action sub-model (referred to as DC model) of Chen's (2020) continuous-time dynamic choice (CTDC) measurement model that defines the easiness parameter only at the task level and ignores the task's process characteristics. The simulation study showed that the SR model performed well in parameter estimation. Moreover, the estimation accuracy of the SR model was quite similar to that of the DC model when state easiness parameters were equal within the task, but was much higher when within-task state easiness parameters were unequal. For the empirical data from the Program for International Student Assessment 2012, the SR model showed better model fit than the DC model. The estimates for state easiness parameters within each task were obviously different and made sense for characterizing task steps, further demonstrating the rationality of the proposed SR model.

Keywords State response model · Process data · Measurement modeling · Problem-solving

With the advances in technology, the use of computers as the delivery platform for assessments facilitates the development of innovative item types, such as simulated interactive tasks (Xiao et al., 2021). Such tasks usually require respondents to interact with the problem scenarios to uncover information, filter and integrate it, and make multistep decisions to approach the solution. Thus, interactive tasks can be used to measure higher-order thinking skills that involve more complex cognitive processes. And this has been put into practice in many large-scale assessments, especially for measuring problem-solving competency, such as the computer-based problem-solving assessments in the Program for International Student Assessment (PISA), the Programme for International Assessment of Adult Competencies (PIAAC), and the National Assessment of Educational Progress (NAEP). One of the typical design frameworks for interactive problem-solving tasks is finite-state automata (FSA) (Buchner & Funke, 1993; Funke, 2001), which have a normative design and easily defined actions and optimal solutions.

For computer-based simulated tasks, a broader range of data can be collected in log files, including not only the final outcomes but also information about how respondents approach the solution (He & von Davier, 2016; Xiao et al., 2021). All the actions of each respondent during their problem-solving process are typically recorded in the form of ordered sequences of multi-type events with timestamps, which can be referred to as the process data. This type of data is valuable when examining interactive tasks (He et al., 2019, 2021). It can promote the understanding of human problem solving, for example, identifying the problem-solving strategies used by respondents and detecting the typical behavioral characteristics of different groups (e.g., Arieli-Attali et al., 2019; He & von Davier, 2015, 2016; Liao et al.,

✉ Hongyun Liu
hyliu@bnu.edu.cn

[1] Department of Educational Psychology, Faculty of Education, East China Normal University, Shanghai, China

[2] Faculty of Psychology, Beijing Normal University, Beijing, China

[3] Beijing Key Laboratory of Applied Experimental Psychology, Faculty of Psychology, Beijing Normal University, Beijing, China

2019; Xiao et al., 2021). More importantly, as the problem-solving process determines the final outcome, process data contain rich information about respondents' problem-solving ability beyond the outcome. Stadler et al. (2020) revealed that individual differences in test-taking behavior sequences indeed indicated differences in problem-solving ability despite the same scores.

However, how to measure individual ability based on process data is a considerable challenge. Unlike traditional test data in which a univariate response is observed for each item, process data are highly unstructured. Specifically, each response process is a sequence of categorical actions (or events). The sequences of different respondents may be completely different, with different lengths and different events that occur at different time points. Moreover, information about the order of actions is critical and should be taken into account in modeling. Therefore, it is difficult to directly apply traditional measurement models to the process data, or even to fully extract meaningful information from it.

To draw valuable inferences from the process data, an increasing number of statistical methods have been proposed in recent years. According to the information obtained, the existing approaches for the analysis of process data can be roughly divided into two categories: (a) methods of extracting features from process data, and (b) measurement models that can infer respondents' latent ability. The first category, feature extraction methods, includes extracting summary statistics according to expert input (e.g., Greiff et al., 2015, 2016), data mining techniques (e.g., He & von Davier, 2015, 2016; Kerr et al., 2011; Liao et al., 2019; Qiao & Jiao, 2018), the use of numerical values or vectors to represent sequences (e.g., the multidimensional scaling approach and the sequence-to-sequence autocoder; Tang et al., 2020, 2021), and so on. This class of methods facilitates the discovery and understanding of problem-solving strategies and behavioral characteristics of respondents. However, these techniques cannot directly provide information about latent ability, and it is difficult to link the obtained features with the latent traits due to a lack of interpretability or theoretical support.

To infer latent traits from process data, some measurement models were proposed, such as the Markov-IRT [item response theory] model (Shu et al., 2017), Markov decision process measurement model (MDP-MM; Lamar, 2018), the modified multilevel mixture IRT model (MMixIRT; Liu et al., 2018), and the continuous-time dynamic choice (CTDC) measurement model (Chen, 2020). These models are built based on action sequences by taking into account the serial dependence in different ways, such as through the Markov property assumption (e.g., Lamar, 2018; Shu et al., 2017), and are somewhat related to traditional measurement models to derive latent trait levels (e.g., Chen, 2020; Lamar, 2018; Liu et al., 2018; Shu et al., 2017).

However, these models have their own limitations. Most of them utilize only limited information about the problem-solving process. For example, in the Markov-IRT model, the transitions between every two adjacent actions are used as indicators and they are scored only based on the frequency of occurrence. Therefore, the sequence order of the response process is not actually preserved in the constructed indicators. In the modified MMixIRT model, the person-level ability estimates are based only on the last step, not the whole process. In addition, these models put much attention on latent abilities, and do not consider or care about task characteristics at the process level that may contribute to understanding the behavioral features of individuals when solving the task.

In this paper, we propose a measurement model for the process data to extract information about both the respondents' latent trait and the task characteristics from the response process. Specifically, we start with FSA tasks, which are commonly used in problem-solving assessments and have been discussed in many studies related to process data analysis methods (e.g., Chen, 2021; Han et al., 2021; Liu et al., 2018; Zhan & Qiao, 2022), and develop the state response (SR) measurement model, which can be applied to process data from one or more tasks. This model focuses on the individual's action choice at each step in the response process. It links these choices with the respondent's latent problem-solving ability and the characteristics of task steps or events, and can be applied to process data from one or more tasks. In addition, the proposed SR model is closely related to the action sub-model of the CTDC model (hereafter referred to as the DC model) that also focuses on the probability of choosing the next action depending on the respondent's latent ability and task parameters. However, the major difference between the two models lies in whether the task characteristics at the process level are taken into account. The DC model only focuses on the overall difficulty of a task, whereas our model goes deeper into each problem state of the task in the problem-solving process.

In the next section, we first briefly describe the FSA tasks and then introduce the proposed model in detail, including the model specification and its estimation, as well as the connection with the related DC model. A simulation study is presented in Section 3 to illustrate the parameter recovery of the proposed model. For comparison, the DC model was also included. Afterward, an empirical study using the real data from PISA 2012 is provided to illustrate the application and rationality of the SR model. Finally, we end this article with a discussion.

State response measurement model

Before clarifying our proposed model, we first briefly introduce the finite-state automata (FSA) tasks. In an FSA task, there are a finite set of system states, a finite set of input

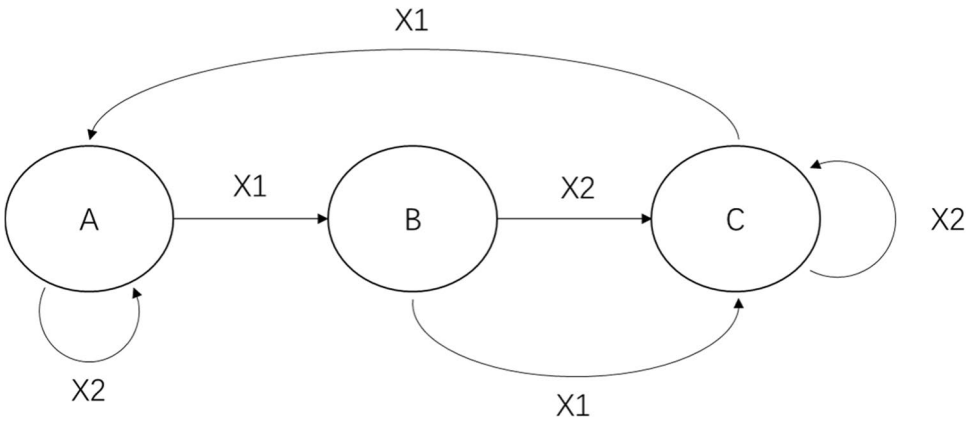

Fig. 1 A graphical representation of an FSA with three states (A, B, C) and two possible actions (X1, X2)

signals (i.e., allowable actions), and a transition function that determines which state will follow from a given state depending on an input signal (Buchner & Funke, 1993; Funke, 2001). Figure 1 presents a graphical representation of an FSA with three states (A, B, C) and two possible actions (X1, X2).

In such tasks, each action can be represented as the resulting state of the problem scenario, which is the cumulative result of system changes caused by all actions that have occurred before. Accordingly, problem states contain part of the information accumulated from the beginning to the current point, and each action sequence can be represented as a corresponding state sequence. In the example of Fig. 1, an action sequence {X1, X1, X2, X1} can be represented as the state sequence {A, B, C, C, A} if the initial problem state is A. A more concrete example can be found in the first task used in the empirical study, for which the problem scenario is described in the "Empirical study" section, and the problem states definition and the state sequence of its optimal solution are provided in Appendix Table 9 and Appendix Fig. 5, respectively (see "Empirical study" section for details).

Model specification

According to the task structure of FSAs introduced above, it can be easily found that when the respondent is in a certain problem state, the reachable states in the next step are a finite set that depends on the current state. In other words, each time the respondent takes an action, they are making a choice among a set of optional events for a certain problem state. According to the problem-solving goal and the events that have occurred before, each choice can be classified as correct or incorrect. Inspired by the idea of IRT modeling, therefore, we view each state as an item and each action choice (i.e., state choice) in the process as a response to the current state, and then model the relationship of the state responses with the characteristics of both the persons and task events. However, in contrast to IRT modeling, which assumes conditional independence between item responses given latent ability, in the proposed model each action choice in the sequence may depend on the previous actions. In addition, a state may appear more than once in a respondent's sequence, unlike the item response data in IRT. The more times the state is visited, the more choice data the respondent produces in that state, and the more information about the state and the person is provided for parameter estimation.

Specifically, the SR model describes the conditional probability of respondent i choosing to reach state s' when they are in problem state s of task k, taking the form

$$P\left(Y_{ik(j+1)} = s' | Y_{ikj} = s, \theta_i, \beta_{ks}, \mathcal{R}\right) = \frac{\exp\left[\left(\beta_{ks} + \theta_i\right) \bullet I_{ssr}\right]}{\sum_{r \in M_s} \exp\left[\left(\beta_{ks} + \theta_i\right) \bullet I_{sr}\right]}, \quad s' \in M_s \tag{1}$$

where Y_{ikj} denotes the jth state in the sequence of respondent i to solve task k; θ_i is the latent ability of respondent i; β_{ks} is the easiness parameter for state s of task k. M_s represents the set of reachable states in the next step given the current state s (which can be understood as optional actions at the current state). The states in M_s can be classified as correct and incorrect according to whether they are closer to the target state given the current situation, and $I_{ss'}$ is an indicator variable that shows the correctness of the reachable state s' when the respondent is in state s. Specifically, if moving from state s to state s' is closer to the target state, $I_{ss'} = 1$; otherwise, $I_{ss'} = 0$. For example, suppose that state A has three reachable states {B, C, D}, in which B is the correct choice and the other two are incorrect choices. The correctness values for reachable states of state A are $I_{AB} = 1$, $I_{AC} = I_{AD} = 0$. Sometimes, this judgment of correctness of reachable states may also depend on previous events in addition to the current state. The problem states of each task, the set of reachable

states for each state, and the correctness of each reachable state need to be predefined manually before data analysis, in which states and their transitions (i.e., their reachable states) always already exist in FSA tasks. These predefined rules of the task(s) are denoted by $\mathcal{R}$. Therefore, in the proposed model, conditional dependencies between actions are taken into account in the form of defined states and correctness of reachable states of each state.

In the model, the state-specific easiness parameter β_{ks} reflects the characteristics of each unique state of the task, showing the propensity to choose a correct next state given the current state s. Since respondents often face the same choices with the same correctness values each time they are in the same state, it is assumed here that each state usually has only one easiness parameter. The state easiness parameter is independent of the latent ability parameter θ_i, and the two parameters jointly determine the probability of a respondent making a choice each time they are in state s. According to Eq. (1), if state s' is a correct choice, the numerator is $\exp(\beta_{ks}+\theta_i)$; otherwise, the numerator is $\exp(0)=1$. The denominator is the sum of the exponential terms of all reachable states for state s, which is used for normalization. Therefore, the larger the β_{ks}, the more likely respondents are to take correct actions in state s in general, thus indicating that state s is easier. Given β_{ks}, the students with a larger value of θ have a higher probability of choosing a correct next state when they are in state s.

Note that in some cases, a state can have more than one easiness parameters, which is related to previously occurring events (i.e., event history). Specifically, although the action options for a state are usually the same each time it is visited, the correctness of those options may sometimes vary according to the information status determined by event history. For example, in the second task of the TICKET unit of PISA 2012 problem-solving assessment (OECD, 2014), students should check the prices of two alternative tickets (ticket 1 and ticket 2) and then buy the cheaper one, i.e., ticket 2. Then, based on the event history, it can be determined at each step which ticket price is already known, resulting in four possible information statuses during the problem-solving process. Suppose that state A is the situation where students are faced with choosing which of the two tickets to check, and the reachable states B and C represent the choice of ticket 1 and ticket 2, respectively. Another reachable state D is the initial state, which means selecting reset in state A to start over. Therefore, when the prices of both tickets are unknown (the first information status), states B and C are both correct, and only state D is incorrect; when the price of only one ticket is known (the second or third information status), it is correct to choose the other ticket (state B or C); and when the prices of two tickets are known (the fourth information status), only state C (i.e., choosing ticket 2) is the correct option and states B and D are both incorrect. Figure 2 shows the transitions from state A with four different sets of correctness. Logically, the easiness of state A may vary across the four cases. However, if we introduce state-history-specific parameters (that is, the easiness of each state

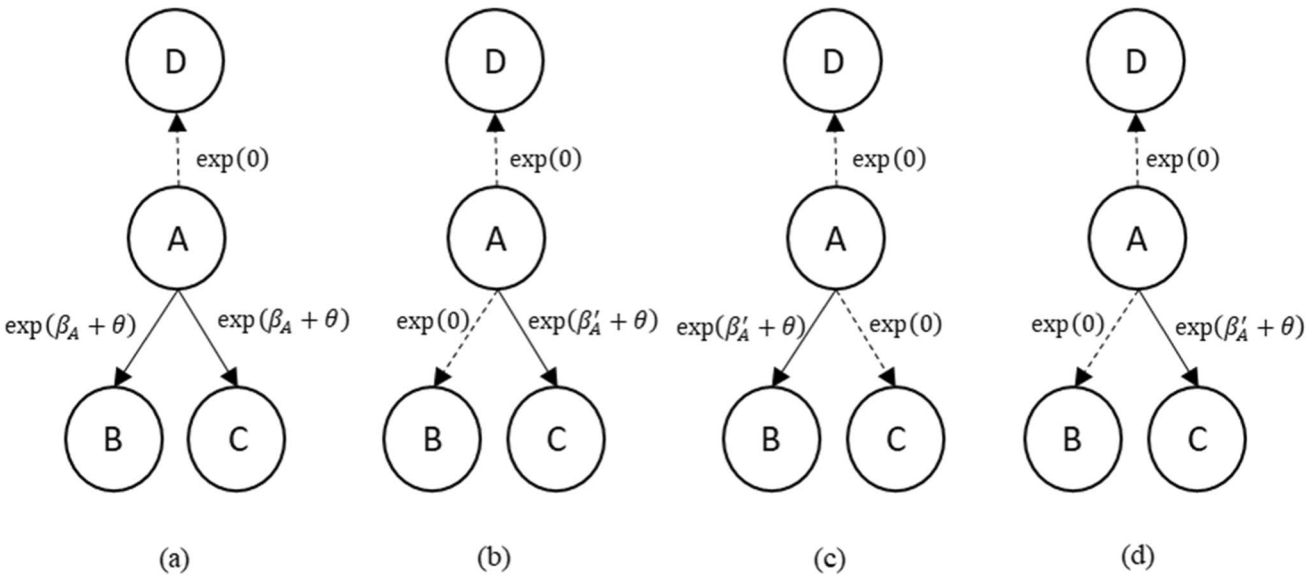

Fig. 2 Transitions from state A with different sets of correctness in four information statuses. Panel ***a*** corresponds to the case where both ticket prices are unknown. Panels ***b*** and ***c*** respectively correspond to the cases where the price of only ticket 1 or 2 is known. Panel ***d*** corresponds to the information status of both ticket prices known. The solid arrow indicates the correct transition, and the dotted arrow represents the incorrect transition. The numerator of the transition probability is annotated beside the corresponding arrow, while the denominator is the sum of numerators across three transitions from state A, for example, $exp(0)+exp(\beta_A+\theta)+exp(\beta_A+\theta)$ in panel (a)

under each history is estimated separately), the model may be much more complex, with a large number of parameters, and the estimation may be poor.

To balance the parsimoniousness and interpretability of the model, for the above cases, the SR model has a simplified assumption. That is, given different event histories (or different information statuses determined by event history), if the number set of correct and incorrect action options for state s remains the same, the easiness parameter for the state (β_s) is assumed to be the same; otherwise, state s given different event histories will be viewed as different states with different easiness parameters. In the above example, when respondents are in state A, there are two correct options and one incorrect option given the first information status, while in the latter three information statuses, there is always one correct option and two incorrect options. Therefore, state A given the first and the latter three event histories will be treated as two different problem states, and their easiness parameters are estimated separately (β_A and β'_A as shown in Fig. 2). In other words, the impact of event history, which can also be considered the temporal dependence, is further incorporated into the current definition of task states.

Denote the sequence length of respondent i in task k as J_{ik}. Assuming the conditional independence between tasks given the latent ability θ_i, the conditional likelihood of action sequences $\boldsymbol{Y}_i = \{\boldsymbol{Y}_{i1}, \boldsymbol{Y}_{ik}, \ldots, \boldsymbol{Y}_{iK}\}$ of respondent i in all K tasks can be written as:

$$\begin{aligned} L(\boldsymbol{Y}_i|\theta_i, \boldsymbol{\beta}_1, \cdots, \boldsymbol{\beta}_K, \mathcal{R}) &= \prod_{k=1}^{K} L(\boldsymbol{Y}_{ik}|\theta_i, \boldsymbol{\beta}_1, \cdots, \boldsymbol{\beta}_K, \mathcal{R}) \\ &= \prod_{k=1}^{K}\prod_{j=1}^{J_{ik}-1} P\left(Y_{ik,(j+1)}\right)\left|Y_{ikj}, \theta_i, \boldsymbol{\beta}_k, \mathcal{R}\right) \end{aligned} \tag{2}$$

where $\boldsymbol{\beta}_k = \left(\beta_{k1}, \ldots, \beta_{kS_k}\right)$ is the vector of easiness parameters of all S_k problem states in task k. For model identification, the mean of θ is set to 0.

If the easiness parameters of all states in the same task are constrained to be equal, we can obtain a simplified version of the proposed SR model. Its form is essentially the same as the action sub-model of the CTDC measurement model of Chen (2020), which is referred to as the DC model. In the DC model, the easiness parameter is task-specific and is the same for all states in a task. Such a specification, however, is unrealistic and restrictive. Since the information for the solution is often gradually revealed in interactive tasks, the difficulty in choosing a correct action given different problem states may be quite different. Therefore, it is conceivable that the DC model ignores the differences between task states and does not probe into the process characteristics of tasks. From this perspective, the DC model is not really built at the process level. By contrast, parameters are constructed for problem states in the proposed SR model, and different unique events in the response process can be distinguished. Then, action choices in different states can provide differentiated information for latent ability estimation. In this sense, the SR model better captures and reflects the dynamics in process data.

Model estimation

In this study, we adopted the Markov chain Monte Carlo (MCMC) method to implement the estimation of the proposed model. The observed data, all sequences of N respondents in K tasks, are denoted as $\boldsymbol{Y}$. The parameters to be estimated include individual latent ability $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_N)$ and state easiness parameters $\boldsymbol{\beta} = (\boldsymbol{\beta}_1, \ldots \boldsymbol{\beta}_k, \ldots, \boldsymbol{\beta}_K)$, in which $\boldsymbol{\beta}_k = \left(\beta_{k1}, \beta_{k2}, \ldots, \beta_{kS_k}\right)$. The joint posterior distribution of interest is

$$p(\boldsymbol{\theta}, \boldsymbol{\beta}|\boldsymbol{Y}, \mathcal{R}) \propto p(\boldsymbol{Y}|\boldsymbol{\theta}, \boldsymbol{\beta}, \mathcal{R}) \cdot p(\boldsymbol{\theta}, \boldsymbol{\beta}), \tag{3}$$

where

$$p(\boldsymbol{Y}|\boldsymbol{\theta}, \boldsymbol{\beta}, \mathcal{R}) = \prod_{i=1}^{N}\prod_{k=1}^{K}\prod_{j=1}^{J_{ik}-1} P\left(Y_{ik,(j+1)}\right)\left|Y_{ikj}, \theta_i, \boldsymbol{\beta}_k, \mathcal{R}\right), \tag{4}$$

$$p(\boldsymbol{\theta}, \boldsymbol{\beta}) = p(\boldsymbol{\theta}) \cdot p(\boldsymbol{\beta}) = \prod_{i=1}^{N} p(\theta_i) \cdot \prod_{k=1}^{k}\prod_{s=1}^{S_k} p(\beta_{ks}). \tag{5}$$

In Eq. (5), $p(\boldsymbol{\theta})$ and $p(\boldsymbol{\beta})$ are the prior distributions of the latent ability and the state parameters, respectively, and they are assumed to be independent of each other. According to the commonly used priors in the MCMC algorithm (e.g., Fox, 2010; Han et al., 2021; Kim & Bolt, 2007; Patz & Junker, 1999b), priors for latent ability and state easiness parameters are set to the standard normal distribution $N(0, 1)$.[1] The initial values for parameters are randomly assigned, yielding the collection of $\boldsymbol{\theta}^0$ and $\boldsymbol{\beta}^0$. The superscript refers to iteration l ($l = 0$ indicating that those are initial values). The Metropolis-Hastings-within-Gibbs sampling approach was used to implement the MCMC estimation to empirically approximate the posterior distributions of parameters (Patz & Junker, 1999a, 1999b). The sampling procedure comprises the following steps for iteration $l+1$:

Step 1. Sample a latent ability θ_i for each respondent. Specifically, draw a candidate value θ_i^* from a proposal distribution centered on the current value θ_i^l, $\theta_i^* \sim N\left(\theta_i^l, \sigma_\theta^2\right)$, independently for each $i = 1, 2, \ldots, N$. Then calculate the acceptance probability for θ_i^*

$$\alpha_i = \min\left\{\frac{p\left(\theta_i^*\middle|\boldsymbol{\beta}^l, \boldsymbol{Y}_i, \mathcal{R}\right)}{p\left(\theta_i^l\middle|\boldsymbol{\beta}^l, \boldsymbol{Y}_i, \mathcal{R}\right)}\right\} = \min\left\{1, \frac{p\left(\boldsymbol{Y}_i\middle|\theta_i^*, \boldsymbol{\beta}^l, \mathcal{R}\right) \cdot p(\theta_i^*)}{p\left(\boldsymbol{Y}_i\middle|\theta_i^l, \boldsymbol{\beta}^l, \mathcal{R}\right) \cdot p(\theta_i^l)}\right\} \tag{6}$$

[1] We also tried to specify a more weakly informative prior for state easiness parameter $\beta_{ks} \sim N(0, 9)$. Results show that the amount of prior information for the state easiness parameters had almost no effect on the parameter estimation of the SR model.

where $p\left(\theta_i^*\right)$ and $p\left(\theta_i^l\right)$ denote the prior probability densities of θ_i^* and θ_i^l, respectively. Draw a random value $r \sim Uniform(0, 1)$. Accept $\theta_i^{l+1} = \theta_i^*$ if $\alpha_i \geq r$; otherwise, $\theta_i^{l+1} = \theta_i^l$.

Step 2. Sample the easiness parameter β_{ks} for each problem state. Draw a candidate value β_{ks}^* from a proposal distribution, $\beta_{ks}^* \sim N\left(\beta_{ks}^l, \sigma_\beta^2\right)$, independently for each $s = 1, 2, \ldots, S_k$ and $k = 1, 2, \ldots, K$. Calculate the acceptance probability

$$\alpha_{ks} = \min\left\{1, \frac{p\left(\beta_{ks}^* \middle| \theta^{l+1}, \boldsymbol{Y}_{ks}, \mathcal{R}\right)}{p\left(\beta_{ks}^l \middle| \theta^{l+1}, \boldsymbol{Y}_{ks}, \mathcal{R}\right)}\right\} = \min\left\{1, \frac{p\left(\boldsymbol{Y}_{ks} \middle| \theta^{l+1}, \beta_{ks}^*, \mathcal{R}\right) \cdot p\left(\beta_{ks}^*\right)}{p\left(\boldsymbol{Y}_{ks} \middle| \theta^{l+1}, \beta_{ks}^l, \mathcal{R}\right) \cdot p\left(\beta_{ks}^l\right)}\right\} \quad (7)$$

where $p\left(\beta_{ks}^*\right)$ and $p\left(\beta_{ks}^l\right)$ denote the prior probability densities of β_{ks}^* and β_{ks}^l, respectively, and $\boldsymbol{Y}_{ks}$ denotes the collection of action choices made by all respondents when they are in state s of task k. Draw a random value $u \sim Uniform(0, 1)$. Set $\beta_{ks}^{l+1} = \beta_{ks}^*$ if $\alpha_{ks} \geq u$; otherwise, $\beta_{ks}^{l+1} = \beta_{ks}^l$.

The variances of proposal distributions, σ_θ^2 and σ_β^2, affect the estimation efficiency and govern the variability in sampling values. In preliminary runs of the chains, the proposal variances σ_θ^2 and σ_β^2 are tuned to control the acceptance rate of each parameter, which is usually between 20% and 60% in practice (Junker et al., 2016; Rosenthal, 2011). Afterward, run multiple chains of length L and then discard a number of initial iterations as burn-in.

The convergence of Markov chains is monitored using the potential scale reduction factor ($\hat{R}$; Brooks & Gelman, 1998; Gelman & Rubin, 1992). The $\hat{R}$ close to 1 indicates that the Markov chains converge to the target distribution.

Simulation study

Design

Three factors were manipulated: (1) sample size (800, 1500, 3000), (2) sequence length (short, medium, long), and (3) the easiness of problem states within task (equal, unequal). The sequence length was mainly controlled by the number of tasks and the number of problem states within each task in the data. Specifically, we simulated two FSA tasks (Task T1, Task T2), involving 9 and 15 problem states, respectively. The task structure, including the problem states, their reachable states, and the corresponding correctness, are listed in Table 1. We then approximated the conditions of short, medium, and long sequence lengths by conducting analyses for Task T1, Task T2, and the two tasks (T1 and T2) together, respectively. Thus, the three levels of sequence length are later represented as Task T1, Task T2, and Two Tasks. In addition, the equal or unequal easiness of problem states within tasks indicates that the true (i.e., the generating) model behind the data was the DC model or the SR model, respectively. The corresponding true values of state easiness parameters when they were unequal within task are listed in Table 1. When the state easiness parameters within task were equal, their true values in tasks T1 and T2 were 1.0 and −0.5, respectively.

In total, we simulated 3*2*3=18 different conditions. For each condition, 50 independent replications were generated based on the corresponding true model. Latent abilities θ_i were drawn from $N(0, 1)$. Each dataset was analyzed

Table 1 Structures of two simulated tasks and the true values of state easiness parameters when they were unequal within each task

Task T1				Task T2			
State	Reachable states		Unequal easiness	State	Reachable states		Unequal easiness
	correct	incorrect			correct	incorrect	
A	B	A	1.103	A	B	A	1.003
B	C	A, G	0.015	B	C	B	0.827
C	D	B, E	0.068	C	D	B, C	0.508
D	I	C, F	0.321	D	E	I	−1.095
E	C	F	−0.536	E	F	C, D	0.517
F	E	I	−0.970	F	G	D, E, L	0.011
G	B	H	−0.564	G	H	E, F, N	0.045
H	G	I	−0.893	H	O	F, G	1.042
				I	D	J	−0.092
				J	D	I, K	−0.559
				K	I	J, O	−0.532
				L	F	M	0.027
				M	F	L, O	−0.441
				N	G	O	−0.392

using the SR and DC models, respectively. The MCMC sampling algorithm for parameter estimation was implemented in R (R Core Team, 2018), in which three chains of 10,000 iterations were used, with the first 2000 iterations discarded as burn-in iterations and every fifth iteration kept. The R code for simulating data and implementing the MCMC algorithm is available at https://osf.io/w9dvf/?view_only=832f623510ba4a7a82ac35fd875e5e30

Note that a few generated sequences were too long due to the randomness of simulation, whereas such sequences were almost impossible in practice and always resulted in negative infinite log-likelihood values in the estimation. To solve this issue, we added a restriction for the input data in the algorithm; that is, only the first 200 problem states in each sequence for each task were used for estimation. The proportion of such sequences that were too long was very low in any generated dataset (the maximum percentage was only 0.75%), and the practice of taking only the first 200 states would not affect the estimation.

Evaluation

Five commonly used indices were applied for model comparison, namely the Akaike information criterion (AIC; Akaike, 1974), Bayesian information criterion (BIC; Schwarz, 1978), the sample size-adjusted BIC (SABIC; Sclove, 1987), deviance information criterion (DIC; Spiegelhalter et al., 1998), and pseudo-Bayes factor (PsBF; Geisser & Eddy, 1979; Gelfand & Dey, 1994). For AIC, BIC, SABIC, and DIC, smaller values indicate a better model fit. PsBF is calculated as the ratio of the conditional predictive ordinates (CPOs) of two models.

$$CPO = \prod_{i=1}^{N} \frac{1}{\frac{1}{R}\sum_{r=1}^{R}\left[p\left(\boldsymbol{x}_i|\Theta^{(r)}\right)\right]^{-1}}, \tag{8}$$

$$PsBF = \frac{CPO(Model\ 1)}{CPO(Model\ 2)}, \tag{9}$$

in which R is the number of MCMC iterations, N denotes the number of persons, $\boldsymbol{x}_i$ denotes the sequence(s) of person i in the data, and $\Theta^{(r)}$ contains values of all parameters to be estimated in the rth iteration. A value of PsBF greater than 3 provides positive (or stronger) evidence in favor of Model 1 and against Model 2 (Levy & Mislevy, 2016, p. 246).

Parameter estimation was evaluated using three criteria: bias and root mean squared error (RMSE) of the estimated values, and their correlations with true values. Note that when evaluating the accuracy of ability estimation, we used the average ability of the same action sequence instead of abilities of single persons.

Results

In all conditions, all parameters successfully converged, of which the $\hat{R}$ values were smaller than 1.1. In the nine conditions with unequal state easiness parameters within task, AIC, BIC, SABIC, DIC, and PsBF all strongly supported the correct SR model across 50 replications, as shown in Table 2. The corresponding estimation accuracy of state easiness and latent ability parameters are shown in the left panels of Figs. 3 and 4, respectively. As seen in Fig. 3, the SR model estimates of state parameters were reasonably accurate under all the simulation settings, which can be shown by negligible average bias and RMSE of less than 0.1. In addition, the estimation accuracy improved with the increase in the sample size. According to the left panel of Fig. 4, the SR estimates of latent ability were acceptable, of which the correlation with true values was higher than 0.8 and the bias was between −0.01 and 0.01. Moreover, a longer sequence length resulted in higher accuracy in the estimation. When comparing the two models, it can be easily observed that the SR model estimation for all parameters was generally more accurate than that DC model estimation, especially for the state easiness parameters.

In the nine conditions of equal state easiness parameters within task, the percentages supporting the DC model across 50 replications are listed in Table 3. As seen from Table 3, only BIC always supported the correct and parsimonious DC model, followed by SABIC, while DIC and PsBF were the least effective. In these conditions, the SR model could still provide good estimation, in which the estimation biases for all parameters were close to zero, RMSE for state parameters was lower than 0.1, and the correlation of ability parameter estimates with true values was higher than 0.9. The estimation accuracy values of the two models for latent ability were very close to each other (see the right panel of Fig. 3). The difference in estimation accuracy of state easiness between the two models was also unsubstantial, although the RMSE for the DC model was slightly smaller (see the right panel of Fig. 4).

Empirical study

Data

Task description

To demonstrate the practical applicability of the proposed SR model, we used data from the first two items of the TICKETS unit in PISA 2012. The problem scenario of the TICKETS unit is an automated ticketing machine, including five interfaces. In the first three interfaces, students can

Table 2 Percentages of replications in which the true SR model was supported in 9 conditions with unequal state easiness parameters within each task

Sample size	Sequence length	AIC	BIC	SABIC	DIC	PsBF
800	Task T1	100%	100%	100%	100%	100%
1500	Task T1	100%	100%	100%	100%	100%
3000	Task T1	100%	100%	100%	100%	100%
800	Task T2	100%	100%	100%	100%	100%
1500	Task T2	100%	100%	100%	100%	100%
3000	Task T2	100%	100%	100%	100%	100%
800	Two Tasks	100%	100%	100%	100%	100%
1500	Two Tasks	100%	100%	100%	100%	100%
3000	Two Tasks	100%	100%	100%	100%	100%

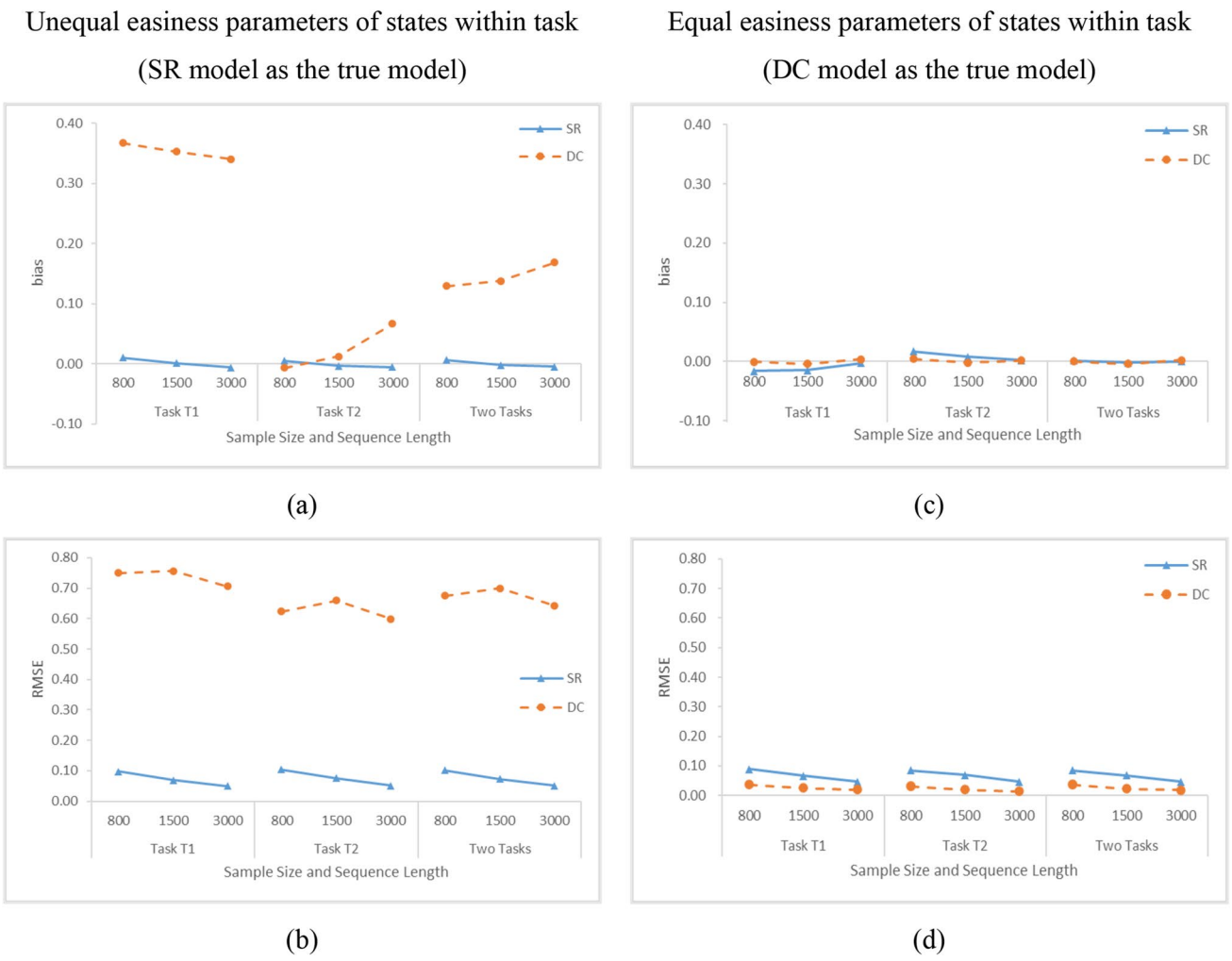

Fig. 3 Estimation accuracy for state easiness parameters using SR and DC models under different conditions in which state easiness parameters within task were unequal (the left panel) or equal (the right panel)

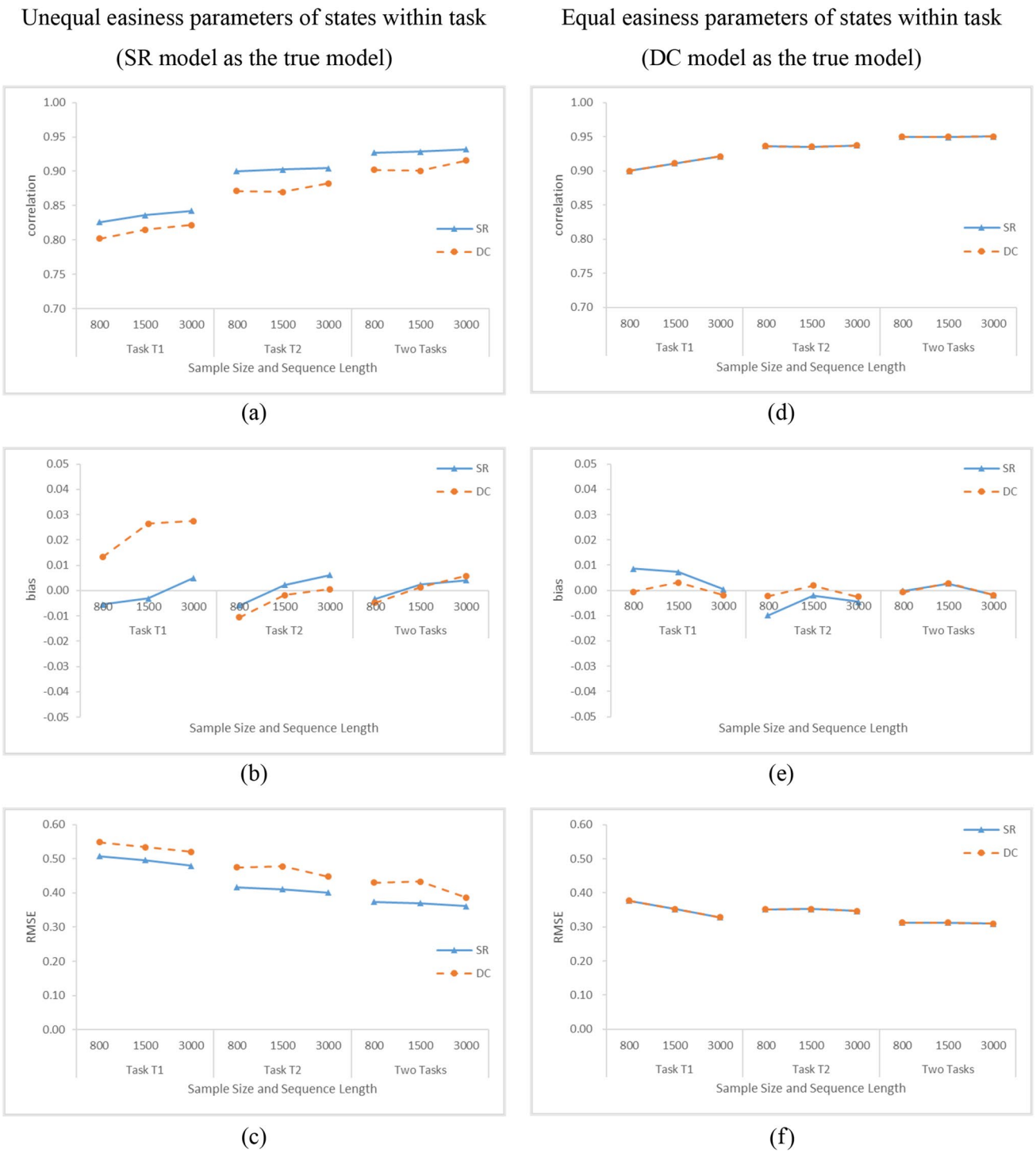

Fig. 4 Estimation accuracy for latent ability using SR and DC models under different conditions in which state easiness parameters within task were unequal (left column) or equal (right column). In panels **d** and **f**, the solid and dashed lines overlap, since the results of the two models are basically the same.

Table 3 Percentages of replications in which the true DC model was supported in 9 conditions with equal state easiness parameters within each task

Sample size	Sequence length	AIC	BIC	SABIC	DIC	PsBF
800	Task T1	92%	100%	98%	80%	58%
1500	Task T1	96%	100%	100%	88%	66%
3000	Task T1	82%	100%	98%	76%	54%
800	Task T2	98%	100%	100%	86%	58%
1500	Task T2	96%	100%	100%	84%	52%
3000	Task T2	84%	100%	100%	58%	58%
800	Two tasks	100%	100%	100%	90%	76%
1500	Two tasks	100%	100%	100%	94%	62%
3000	Two tasks	98%	100%	100%	82%	66%

choose the train network (CITY SUBWAY, COUNTRY TRAINS, CANCEL), fare type (FULL FARE, CONCESSION, CANCEL), and ticket type (DAILY, INDIVIDUAL, CANCEL) in order. If a student selects DAILY, the next interface will show the price of the selected ticket, and two options, BUY or CANCEL. Alternatively, if the student selects INDIVIDUAL, the next interface will show the available number of individual trips (1 to 5), as well as BUY and CANCEL buttons. After the student selects a certain number, the price of the ticket will be presented. When the student clicks BUY, the task terminates. The CANCEL button in each interface allows the student to reset all choices and navigate to the initial interface. More details about the unit can be found in the PISA 2012 results report (OECD, 2014).

The first item of the TICKETS unit required students to buy a full-fare, country train ticket with two individual trips. The requirements for the ticket were very clear, and students only needed to make choices on the machine following those requirements. The optimal solution was to select the network "COUNTRY TRAINS," the fare type "FULL FARE," the ticket type "INDIVIDUAL," and the number of tickets "2" in that order, and finally click BUY. This item was dichotomously scored based on whether the student purchased the correct ticket.

The second item was more complicated. Students were asked to find and buy the cheapest ticket that allowed them to take four trips around the city on the subway within a day, and they were told that they could use concession fares. To complete this task, students had to find and compare the prices of two possible alternatives that satisfied the ticket requirements, which were a daily subway ticket with concession fare, and an individual concession fare subway ticket with four trips. Afterward, the student had to purchase the cheaper ticket, which was the individual concession fare subway ticket with four trips. In PISA 2012, this task was polytomously scored as 0/1/2. Only if the student compared the two prices and purchased the correct ticket would they be considered to have successfully solved the task and receive full credit. If the student purchased one of the two tickets without comparing prices, they could be given only partial credit.

The raw process data and item scores of the two tasks are available from the OECD website: http://www.oecd.org/pisa/pisaproducts/database-cbapisa2012.htm. Students' process data were organized into state sequences according to the definition of problem states.

Definitions of problem states and correctness

All the problem states for the two tasks, their reachable states, and corresponding correctness are provided in the Appendix Tables 9 to 11, which are the same as in Chen (2020). For an intuitive understanding, screenshots of the optimal solution for the first task, as well as the corresponding defined problem states, are provided in Appendix Fig. 5 as an example.

Note that in the second task, the correct and incorrect options of states S7, S9, S10, and S11 vary with the information status caused by previous actions (see Appendix Table 11). For example, S7 represents the case in which the participant faces the choice of ticket type after choosing the correct network (CITY SUBWAY) and the correct fare type (CONCESSION). If the participant does not know the prices of two tickets that meet the travel requirement (information status A), both ticket types are correct choices. If the price of one of the two tickets has been known (information status B or C), another ticket type is the only correct choice. And if the prices of both tickets are known (information status D), only the INDIVIDUAL ticket type is correct. According to the simplified assumption mentioned in the *Model specification* section, states S9, S10, and S11 have one correct option and three incorrect options in all information statuses, and therefore each of them is considered to have only one easiness parameter regardless of the information status. By contrast, given the information status A, the numbers of

correct and incorrect options for S7 are 2 and 1, respectively, while the numbers are 1 and 2 given the other three information statuses B–D. Accordingly, S7, given the information status A and B–D, was treated as two different states S7_1 and S7_2, respectively, of which the state parameters were estimated separately.

Sample

After data cleaning, the sequences of 27,616 students who completed the two tasks were used for process data analysis. The sequence length of the first task ranged from 5 to 146, with a mean of 7.39 and median of 6. The sequence length of the second task ranged from 5 to 91, with a mean of 10.05 and median of 6.

After the process data analysis, we examined the ability estimates obtained, in which item scores were used. Since a small number of students had missing data in the scores of one or both tasks, after matching the item scores and ability estimates, the data of only 26,718 students were included in this stage.

Analysis

We applied both the SR model and the DC model to the process data from the two tasks. According to the findings of the previous simulation study, longer sequences contribute to better estimation of the latent trait. Therefore, sequences from the two tasks were analyzed together. In the SR and DC models, the priors of the latent ability and state easiness parameters were specified as $N(0, 1)$. Model fit indices AIC, BIC, ABIC, DIC, and PsBF were used for model comparison.

The latent ability estimates from two models were compared by their correlations with the task outcome scores and their explanatory power to the overall problem-solving performance in PISA 2012. That is, we regressed the overall performance scores on the ability estimates from different models and compared the R^2 values. In PISA 2012, the plausible values are a selection of likely proficiencies for students based on the scores of tasks they received, and five plausible values were generated for each student (OECD, 2014). Following Greiff et al. (2015) and Chen (2020), the first plausible value of problem-solving proficiency provided in PISA 2012 products was used as the overall performance score.

Results

In the analysis of two tasks, both models successfully converged, as the potential scale reduction factor values for all parameters were between 1 and 1.01. Table 4 lists the model fit for the two models. All five indices strongly supported the SR model over the DC model. Therefore, the easiness parameters of the problem states within each task might be quite different and should not be fixed to be equal.

The state parameter estimates for the two tasks are presented in Tables 5 and 6. The easiness parameters of states in the same task were quite different and their 95% credible intervals had almost no overlap, which was consistent with the model comparison results. It can be seen from Tables 5 and 6 that the easiness parameters of states in the optimal solution were generally higher than those of other states. For example, S16 in the first task had the highest estimated value (2.578). This implies that the respondents were very likely to directly click the "BUY" button when they arrived at the ticket purchase interface after choosing the correct network (COUNTRY TRAIN), the correct fare type (FULL), the correct ticket type (INDIVIDUAL), and the correct number of individual trips (2). S14 in the first task also had a high easiness estimate (2.575), which indicates that the respondents who had selected the correct network (COUNTRY TRAIN), correct fare type (FULL), and correct ticket type (INDIVIDUAL) were very likely to choose the correct number of individual trips. In the second task, S7_1 had the highest easiness estimate (3.590). This state indicates that the respondents who had correctly selected CITY SUBWAY and CONCESSION fare needed to choose a ticket type (INDIVIDUAL or DAILY) when they did not know the prices of the two tickets that met the trip requirement. Therefore, it should be very easy to make the right response, as both ticket types were the right choices at that time. By contrast, the estimated value of S7_2 was lower (1.318), which means that after obtaining the price information of one or two tickets that met the trip requirement, students needed to

Table 4 Model fit of two models in the empirical study

Model	AIC	BIC	ABIC	DIC	2ln(PsBF)
SR	595,405.04	595,742.29	595,612.06	612,796.85	87,921.69
DC	671,978.74	671,995.17	671,988.88	691,163.49	

When PsBF was calculated using Eq. (9), the SR model was used as Model 1 and the DC model was used as Model 2. Additionally, for numerical stability, we worked with logarithms when calculating PsBF, and present the results in natural logarithmic form.

Table 5 Estimates of state easiness parameters of the first task in the empirical study

State	SR		DC	
	Mean	95% CI	Mean	95% CI
S1	**1.303**	**[1.278, 1.328]**	1.231	[1.215, 1.249]
S2	−0.848	[−0.903, −0.792]		
S3	−1.409	[−1.507, −1.317]		
S4	−0.379	[−0.461, −0.299]		
S5	−1.087	[−1.230, −0.945]		
S6	−0.636	[−0.752, −0.529]		
S7	−1.317	[−1.438, −1.200]		
S8	−0.391	[−0.490, −0.295]		
S9	−0.490	[−0.648, −0.331]		
S10	−0.697	[−0.848, −0.552]		
S11	**2.361**	**[2.323, 2.399]**		
S12	**2.191**	**[2.151, 2.230]**		
S13	−0.653	[−0.738, −0.570]		
S14	**2.575**	**[2.530, 2.621]**		
S15	0.932	[0.852, 1.010]		
S16	**2.578**	**[2.532, 2.624]**		
S17	−0.795	[−0.897, −0.691]		
S18	−0.332	[−0.477, −0.195]		
S19	−0.642	[−0.779, −0.505]		
S20	−0.649	[−0.781, −0.523]		

CI = credibility interval. The states in the optimal solution are shown in bold.

Table 6 Estimates of state easiness parameters of the second task in the empirical study

State	SR		DC	
	Mean	95% CI	Mean	95% CI
S1	**2.192**	**[2.167, 2.218]**	1.454	[1.438, 1.470]
S2	**2.887**	**[2.853, 2.922]**		
S3	−0.454	[−0.538, −0.370]		
S4	0.629	[0.537, 0.720]		
S5	−1.267	[−1.477, −1.067]		
S6	0.785	[0.660, 0.908]		
S7_1	**3.590**	**[3.492, 3.697]**		
S7_2	**1.318**	**[1.278, 1.358]**		
S8	**0.107**	**[0.074, 0.141]**		
S9	**1.886**	**[1.85, 1.922]**		
S10	−0.030	[−0.091, 0.030]		
S11	**0.581**	**[0.547, 0.615]**		
S12	−0.210	[−0.265, −0.154]		
S13	−0.776	[−0.958, −0.596]		
S14	0.387	[0.230, 0.549]		
S15	−0.966	[−1.286, −0.662]		
S16	0.218	[0.032, 0.404]		
S17	−1.692	[−1.809, −1.573]		
S18	0.858	[0.760, 0.956]		
S19	−1.413	[−1.558, −1.263]		
S20	0.831	[0.759, 0.906]		

CI = credibility interval. The states in the optimal solution are shown in bold.

be more careful to correctly choose another ticket type to check its price or to correctly choose the cheaper of the two tickets.

By contrast, the DC model provided only task-level parameters, of which the information was limited. In addition, it is counterintuitive that the easiness of the second task was higher than that of the first task. Chen (2020) speculated that the familiarity with the task interface was also included in the task-level easiness parameter and that it was not difficult to partially solve the problem in the second task, thus reducing the task's overall difficulty. Nevertheless, this result is still difficult to understand.

As for the latent ability, the estimates provided by the SR model and the DC model were highly consistent, with correlation coefficients of 0.977. The correlations between the ability estimates and the outcome scores are given in Table 7. For the first task, the difference between the correlation values of the two models was not substantial. However, for the second task, the SR model estimates were more strongly correlated with the task outcome than those of the DC model.

We further compared the R^2 values of regressions of individuals' overall performance on the ability estimates obtained by the two process data analysis models. The two regression models were significant ($p < 0.01$), and the corresponding estimation results are shown in Table 8. The slope parameters were significantly positive, indicating that students with higher process-based estimates tended to have better overall performance in problem solving, which is in line with expectations. Further, results show that the SR model estimates had higher explanatory power of the overall performance ($R^2 = 0.384$) than the DC model estimates ($R^2 = 0.361$). This implies that, due to the considerations of process steps, the SR model estimates for latent ability are more informative about the individuals' overall problem-solving competence than the DC model estimates.

Table 7 Correlations between ability estimates from two models and task outcomes

Model	Task 1	Task 2
SR	0.769^{**}	0.785^{**}
DC	0.792^{**}	0.719^{**}

$^{**}p < 0.01$.

Table 8 Estimation results for two regression models that regress the overall performance score on ability estimates from the SR model (M_1) and DC model (M_2) in the empirical study

	Coefficients	Unstandardized coefficients		Standardized coefficients	*t*	*p*	R^2
		Estimate	*SE*				
M_1	(Constant)	497.468	0.483		1030.629	0.000	0.384
	θ_{SR}	74.158	0.574	0.620	129.121	0.000	
M_2	(Constant)	497.445	0.492		1011.872	0.000	0.361
	θ_{DC}	56.458	0.459	0.601	122.919	0.000	

Discussion

Different people usually react differently to the same task, resulting in a variety of action sequences. These sequences always contain richer information than the outcomes, not only about the respondents, but also about the tasks. In this study, starting with FSA tasks, we develop a state response measurement model for problem-solving process data, which is a discrete choice model and can reflect the characteristics of both persons and task steps. Through the predefined correctness of events that are available as the next action, the SR model links the action choice with the latent ability of the respondent and the easiness of the current problem state. Results of the simulation study show that the proposed SR model could provide a reasonably accurate estimation of parameters regardless of whether the state easiness parameters were indeed equal within tasks. Longer sequences (or more tasks) helped to improve the estimation accuracy of ability parameters, and a larger sample size contributed to a better estimation of state parameters.

The proposed SR model was also applied to the process data from two problem-solving tasks in PISA 2012. For each problem state, an estimate of its easiness was obtained, and the value made sense for characterizing the corresponding task step. In addition, SR model estimates for ability parameters explained nearly 40% of the variance in students' overall performance scores reported by PISA 2012 and had a certain degree of correlation with the outcome scores of the two tasks.

In both simulation and empirical studies, we also included the DC model—i.e., the action sub-model of Chen's (2020) CTDC model—for comparison. The DC model can be viewed as a special case of our proposed SR model that constrains the easiness of all states in the same task to be equal. Accordingly, the easiness parameters related to tasks in the model are task-specific, not state-specific. This constraint on task states is unrealistic and ignores the task characteristics at the process level. As shown in our simulation study, the task easiness parameters in the DC model provided limited and possibly inaccurate information about tasks. However, the proposed SR model overcomes this disadvantage of the DC model. The state-specific parameters included in the SR model reflect the process features of each task, that is, the difficulty of different task states (or steps). Such specification is closer to reality, and the estimation results are more accurate and informative. As shown in the empirical study, due to the consideration of task states in the SR model, its ability estimates contain more information about the overall performance than the DC model estimates. In addition, the estimates are more consistent with the outcome scores of the more complex second task, which were given with partial consideration of the response process.

Above all, the proposed model provides an effective measurement framework for analyzing process data. It can reveal information on both the person and task aspects from the process data. In fact, most of the existing research on process data focuses primarily on the person-level information or characteristics, such as the strategies used, the types of mistakes, and the latent traits (e.g., Shu et al., 2017; Stadler et al., 2020; Tang et al., 2020; Zhan & Qiao, 2022). By contrast, the proposed model not only can provide the estimation of individual ability, but also considers the process characteristics of tasks, that is, the difficulty of each step in the task, which can aid in understanding the interactive problem-solving tasks and individuals' behavioral characteristics. Parameter estimates of problem states may also provide the item designers and researchers in the field of cognition with more specific directions for task improvement. In addition, the SR model can handle data with different types of missingness. Since it focuses on the action choice in each state, and the dependence between states is included in the model through the predefined correctness, the SR model can be applied normally when some participants complete only a subset of tasks due to test design, or when individuals' actions after a certain time point are not observed, for example, due to time limits.

Although the SR model is constructed based on the FSA tasks in this paper, the model application is not limited to this type of task. Actually, the key to using the SR model lies in the predefinition of all problem states, the relations between them (that is, the optional next states of each state), and their correctness as the next state. In FSA tasks, problem states and the transitions between them are built into the task design, and thus researchers

usually only need to define the correctness of the reachable states for each state. For other types of tasks without built-in problem states, the data preprocessing work, such as the definition of problem states and their correctness and data recoding, can be implemented manually, which requires the involvement of content experts. In this step, some data-driven algorithms (e.g., hidden Markov modeling) can be additionally considered to provide information about problem-solving sub-phases, thereby assisting in the identification of problem states.

Limitations and future directions

Despite its flexibility, the proposed model has some limitations that remain to be improved in the future. First, although the correctness value of each reachable state in the SR model is dichotomous (0 or 1) in this paper, it can be defined as a value between 0 and 1 (or other lower and upper limits) and can be different for different reachable states, indicating the efficiency of choosing different next actions for achieving the target state. In future research, this correctness can also be included as a parameter to be estimated in the model, similar to the reward function in Lamar's (2018) Markov decision process measurement model.

Second, when the correct action choice of a state varies with event history, we treat this state given different information statuses caused by event history as different states based on the number of its correct and incorrect options in this study. This is partly due to the consideration of parameter estimation. The model already includes a number of parameters since it considers each task state. Further introducing state-history-specific parameters may result in poor model performance. However, the state easiness is likely to be related to the specific correct and incorrect events in that state. Additionally, in the proposed model, we assume that all parameters are usually static. In other words, the state easiness and students' ability parameters remain constant during the whole problem-solving process. This assumption may hold for relatively simple tasks without feedback, such as the TICKET tasks used in this study. However, for more complex dynamic interactive tasks, respondents may receive feedback from the task scenarios, resulting in an increase in their ability, and a state may become easier after the respondent visits the state several times. Therefore, determining a way to consider the influence of the previous events in the model more reasonably is an interesting issue that needs careful consideration.

Third, some states may be rarely reached if there are many allowable actions in a task that lead to many possible states. In such cases, the easiness parameters for these states with few response data may not be stably estimated. For this issue, one way is to reduce the number of states in the predefinition stage. For example, some unimportant states can be combined into a more general state. Solving this issue from the perspective of model estimation can also be considered. Specifically, parameter estimation methods for IRT models dealing with sparse response matrices, small samples, and missing data, such as regularized estimation (Battauz, 2020; Chen et al., 2021), can be introduced and adapted to the current model framework, or some improvements to the current Bayesian estimation procedure used in this paper can be attempted along the lines of these methods, such as the use of hierarchical priors (e.g., Gilholm et al. 2021; König et al. 2020).

Conclusions

In this study, we propose a new SR measurement model for process data analysis by incorporating the characteristics of action sequences and the concept of IRT modeling. The SR model takes full advantage of the whole solution sequence by focusing on the action choice at each response step, and takes into account the temporal dependence in the sequence by predefining the correctness of each choice. The application of the SR model holds promise in providing deeper insights into individuals' behavioral characteristics in interactive tasks and their latent ability levels, and equally importantly, it offers a new perspective for understanding interactive tasks, which can be helpful in designing, evaluating, and improving new types of technology-based assessments with interactive modes. Overall, the SR model provides an analytical framework with great potential for process data in computer-based interactive tasks.

Appendix

Table 9 A list of 21 problem states for the first item of the TICKETS unit, the corresponding reachable states and the correctness

State	Network	Fare type	Ticket type	Number of individual trips	End	Reachable states	
						Correct	Incorrect
S1	NULL	NULL	NULL	NULL	0	S11	S1, S2
S2	CITY SUBWAY	NULL	NULL	NULL	0	S1	S3, S7
S3	CITY SUBWAY	FULL	NULL	NULL	0	S1	S4, S5
S4	CITY SUBWAY	FULL	DAILY	NULL	0	S1	S21
S5	CITY SUBWAY	FULL	INDIVIDUAL	NULL	0	S1	S6, S21
S6	CITY SUBWAY	FULL	INDIVIDUAL	1/2/3/4/5	0	S1	S6, S21
S7	CITY SUBWAY	CONCESSION	NULL	NULL	0	S1	S8, S9
S8	CITY SUBWAY	CONCESSION	DAILY	NULL	0	S1	S21
S9	CITY SUBWAY	CONCESSION	INDIVIDUAL	NULL	0	S1	S10, S21
S10	CITY SUBWAY	CONCESSION	INDIVIDUAL	1/2/3/4/5	0	S1	S10, S21
S11	COUNTRY TRAIN	NULL	NULL	NULL	0	S12	S1, S17
S12	COUNTRY TRAIN	FULL	NULL	NULL	0	S14	S1, S13
S13	COUNTRY TRAIN	FULL	DAILY	NULL	0	S1	S21
S14	COUNTRY TRAIN	FULL	INDIVIDUAL	NULL	0	S16	S1, S15, S21
S15	COUNTRY TRAIN	FULL	INDIVIDUAL	1/3/4/5	0	S16	S1, S15, S21
S16	COUNTRY TRAIN	FULL	INDIVIDUAL	2	0	S21	S1, S15, S16
S17	COUNTRY TRAIN	CONCESSION	NULL	NULL	0	S1	S18, S19
S18	COUNTRY TRAIN	CONCESSION	DAILY	NULL	0	S1	S21
S19	COUNTRY TRAIN	CONCESSION	INDIVIDUAL	NULL	0	S1	S20, S21
S20	COUNTRY TRAIN	CONCESSION	INDIVIDUAL	1/2/3/4/5	0	S1	S20, S21
S21	NULL	NULL	NULL	NULL	1	—	—

Each row represents a problem state, denoting the choice in "network," "fare type," "ticket type," and "number of individual trips." The value of "NULL" means no choice has been made for the corresponding ticket condition. For example, S2 denotes the state where the respondent has chosen CITY SUBWAY for the traffic network and does not choose the other ticket conditions (i.e., fare type, ticket type, and number of individual trips). "End" implies whether the task is over. S21 is the end state and so it has no reachable states.

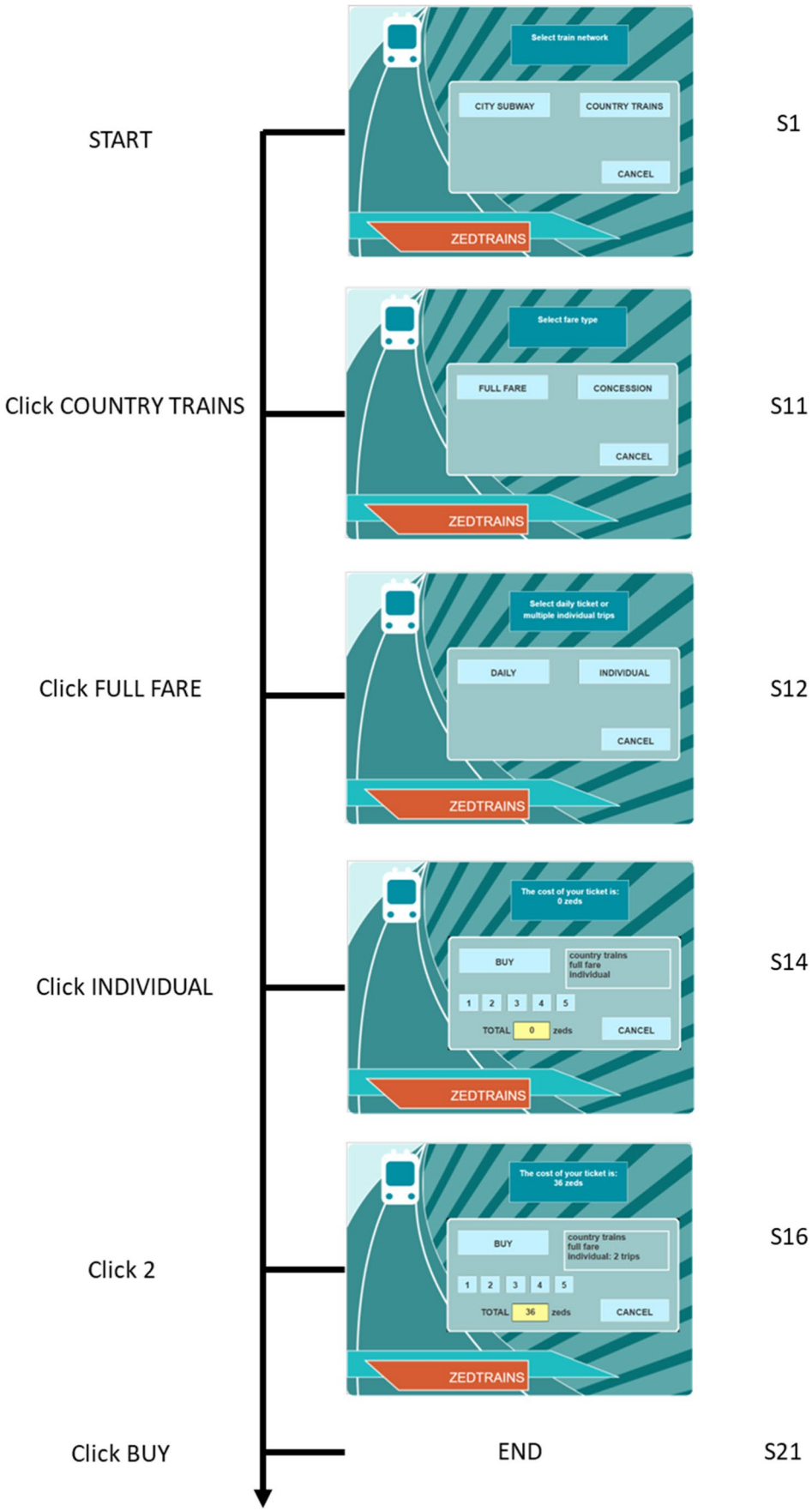

Fig. 5 Screenshots of the optimal solution and the corresponding problem states for the first item of the TICKETS unit. (For a clearer view, please see http://www.oecd.org/pisa/test-2012/testquestions/question4/).

Table 10 A list of 21 problem states for the second item of the TICKETS unit

State	Network	Fare type	Ticket type	Number of individual trips	End
S1	NULL	NULL	NULL	NULL	0
S2	CITY SUBWAY	NULL	NULL	NULL	0
S3	CITY SUBWAY	FULL	NULL	NULL	0
S4	CITY SUBWAY	FULL	DAILY	NULL	0
S5	CITY SUBWAY	FULL	INDIVIDUAL	NULL	0
S6	CITY SUBWAY	FULL	INDIVIDUAL	1/2/3/4/5	0
S7	CITY SUBWAY	CONCESSION	NULL	NULL	0
S8	CITY SUBWAY	CONCESSION	DAILY	NULL	0
S9	CITY SUBWAY	CONCESSION	INDIVIDUAL	NULL	0
S10	CITY SUBWAY	CONCESSION	INDIVIDUAL	1/2/3/5	0
S11	CITY SUBWAY	CONCESSION	INDIVIDUAL	4	0
S12	COUNTRY TRAIN	NULL	NULL	NULL	0
S13	COUNTRY TRAIN	FULL	NULL	NULL	0
S14	COUNTRY TRAIN	FULL	DAILY	NULL	0
S15	COUNTRY TRAIN	FULL	INDIVIDUAL	NULL	0
S16	COUNTRY TRAIN	FULL	INDIVIDUAL	1/2/3/4/5	0
S17	COUNTRY TRAIN	CONCESSION	NULL	NULL	0
S18	COUNTRY TRAIN	CONCESSION	DAILY	NULL	0
S19	COUNTRY TRAIN	CONCESSION	INDIVIDUAL	NULL	0
S20	COUNTRY TRAIN	CONCESSION	INDIVIDUAL	1/2/3/4/5	0
S21	NULL	NULL	NULL	NULL	1

Each row represents a problem state, denoting the choice in "network," "fare type," "ticket type," and "number of individual trips." The value of "NULL" means no choice has been made for the corresponding ticket condition. For example, S2 denotes the state that the respondent has chosen CITY SUBWAY for the traffic network and does not choose the other ticket conditions (i.e., fare type, ticket type, and number of individual trips). "End" implies whether the task is over.

Table 11 The reachable states and the corresponding correctness given each problem state for the second item of the TICKETS unit

State	Information status A		Information status B		Information status C		Information status D	
	Correct	Incorrect	Correct	Incorrect	Correct	Incorrect	Correct	Incorrect
S1	S2	S1, S2	S2	S1, S2	S2	S1, S2	S2	S1, S2
S2	S7	S1, S3	S7	S1, S3	S7	S1, S3	S7	S1, S3
S3	S1	S4, S5	S1	S4, S5	S1	S4, S5	S1	S4, S5
S4	S1	S21	S1	S21	S1	S21	S1	S21
S5	S1	S6, S21	S1	S6, S21	S1	S6, S21	S1	S6, S21
S6	S1	S6, S21	S1	S6, S21	S1	S6, S21	S1	S6, S21
S7	**S8, S9**	**S1**	**S9**	**S1, S8**	**S8**	**S1, S9**	**S9**	**S1, S8**
S8	S1	S21	S1	S21	S1	S21	S1	S21
S9	**S11**	**S1, S10, S21**	**S11**	**S1, S10, S21**	**S1**	**S10, S11, S21**	**S11**	**S1, S10, S21**
S10	**S11**	**S1, S10, S21**	**S11**	**S1, S10, S21**	**S1**	**S10, S11, S21**	**S11**	**S1, S10, S21**
S11	**S1**	**S10, S11, S21**	**S21**	**S1, S10, S11**	**S1**	**S10, S11, S21**	**S21**	**S1, S10, S11**
S12	S1	S13, S17	S1	S13, S17	S1	S13, S17	S1	S13, S17
S13	S1	S14, S15	S1	S14, S15	S1	S14, S15	S1	S14, S15
S14	S1	S21	S1	S21	S1	S21	S1	S21
S15	S1	S16, S21	S1	S16, S21	S1	S16, S21	S1	S16, S21
S16	S1	S16, S21	S1	S16, S21	S1	S16, S21	S1	S16, S21
S17	S1	S18, S19	S1	S18, S19	S1	S18, S19	S1	S18, S19
S18	S1	S21	S1	S21	S1	S21	S1	S21
S19	S1	S20, S21	S1	S20, S21	S1	S20, S21	S1	S20, S21
S20	S1	S20, S21	S1	S20, S21	S1	S20, S21	S1	S20, S21
S21	—	—	—	—	—	—	—	—

Since this task requires comparison before making a decision, the previous actions lead to different information statuses about ticket prices, thus affecting the correct and incorrect reachable states of some states. According to the task requirement, four information statuses are defined to indicate whether the fare of a concession daily subway ticket and/or the fare of an individual concession subway ticket with four trips are known. Information status A means that the fares of the two tickets are unknown, which is determined by the absence of S8 and S11 in the previous actions. Information status B indicates that the fare of a concession daily subway ticket is known but the other ticket fare is unknown, which is operationalized as the absence of S11 but the existence of S8 in the previous actions. Information status C denotes that the fare of the individual concession subway tickets with four trips is known while the other fare is unknown, which is determined by the absence of S8 but the existence of S11 in the previous states. Information status D means that the fares of the two tickets are known; that is, both S8 and S11 have appeared in the previous states. The states whose correct and incorrect options vary with the information status are displayed in bold.

Acknowledgements The authors thank the OECD-PISA (Program for International Student Assessment) team for granting access to the data source and instruments in this study.

Author contributions Yue Xiao contributed to the conceptualization of the study, performed the analysis and wrote the manuscript. Hongyun Liu contributed to the supervision of the study and helped revise the manuscript with constructive discussions.

Funding This work was supported by National Natural Science Foundation of China (32071091).

Data Availability The data that support the findings of this study are openly available from the OECD website: http://www.oecd.org/pisa/pisaproducts/database-cbapisa2012.htm.

Declarations

Conflict of Interest The authors declare that they have no known conflict of interest to disclose.

References

Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control, 19*, 716–723.

Arieli-Attali, M., Ou, L., & Simmering, V. R. (2019). Understanding test Takers' choices in a self-adapted test: A hidden Markov modeling of process data. *Frontiers in Psychology, 10*, Article 83. https://doi.org/10.3389/fpsyg.2019.00083

Battauz, M. (2020). Regularized estimation of the nominal response model. *Multivariate Behavioral Research, 55*(6), 811–824. https://doi.org/10.1080/00273171.2019.1681252

Brooks, S. P., & Gelman, A. (1998). General methods for monitoring convergence of iterative simulations. *Journal of Computational and Graphical Statistics, 7*(4), 434–455.

Buchner, A., & Funke, J. (1993). Finite-state automata: Dynamic task environments in problem-solving research. *The Quarterly Journal of Experimental Psychology, 46*(1), 83–118.

Chen, Y. (2020). A continuous-time dynamic choice measurement model for problem-solving process data. *Psychometrika, 85*(4), 1052–1075. https://doi.org/10.1007/s11336-020-09734-1

Chen, Y., Li, X., Liu, J., & Ying, Z. (2021). *Item Response Theory — A Statistical Framework for Educational and Psychological Measurement*. arXiv. https://arxiv.org/abs/2108.08604

Funke, J. (2001). Dynamic systems as tools for analysing human judgement. *Thinking & Reasoning, 7*, 69–89.

Fox, J. P. (2010). *Bayesian Item Response Modeling: Theory and Application*. Springer.

Gelman, A., & Rubin, D. B. (1992). Inference from iterative simulation using multiple sequences. *Statistical Science, 7*, 457–511.

Geisser, S., & Eddy, W. F. (1979). A predictive approach to model selection. *Journal of the American Statistical Association, 74*, 153–160.

Gelfand, A. E., & Dey, D. K. (1994). Bayesian model choice: Asymptotics and exact calculations. *Journal of the Royal Statistical Society, Series B, 56*, 501–514.

Gilholm, P., Mengersen, K., & Thompson, H. (2021). Bayesian Hierarchical Multidimensional Item Response Modeling of Small Sample, Sparse Data for Personalized Developmental Surveillance. *Educational and Psychological Measurement, 81*(5), 936–956. https://doi.org/10.1177/0013164420987582

Greiff, S., Wüstenberg, S., & Avvisati, F. (2015). Computer-generated log-file analyses as a window into students' minds? A showcase study based on the PISA 2012 assessment of problem solving. *Computers & Education, 91*, 92–105.

Greiff, S., Niepel, C., Scherer, R., & Martin, R. (2016). Understanding students' performance in a computer-based assessment of complex problem solving: An analysis of behavioral data from computer-generated log files. *Computers in Human Behavior, 61*, 36–46.

Han, Y., Liu, H., & Ji, F. (2021). A Sequential Response Model for Analyzing Process Data on Technology-Based Problem-Solving Tasks. *Multivariate Behavioral Research*. Advance Online Publication. http://doi.org/https://doi.org/10.1080/00273171.2021.1932403

He, Q., Borgonovi, F., & Paccagnella, M. (2019). *Using Process Data to Understand Adults' Problem-Solving Behaviour in the Programme for the International Assessment of Adult Competencies (PIAAC): Identifying Generalised Patterns Across Multiple Tasks with Sequence Mining*. OECD Education Working Papers (OECD Publishing). https://doi.org/10.1787/650918f2-en

He, Q., Borgonovi, F., & Paccagnella, M. (2021). Leveraging process data to assess adults' problem-solving skills: Identifying generalized behavioral patterns with sequence mining. *Computers & Education, 166*, Article 104170. https://doi.org/10.1016/j.compedu.2021.104170

He, Q., & von Davier, M. (2015). Identifying feature sequences from process data in problem-solving items with n-grams. In L. A. van der Ark, D. M. Bolt, W.-C. Wang, J. A. Douglas, & S.-M. Chow (Eds.), *Quantitative psychology research* (pp. 173–190). Springer. https://doi.org/10.1007/978-3-319-19977-1_13

He, Q., & von Davier, M. (2016). Analyzing process data from problem-solving items with ngrams: Insights from a computer-based large-scale assessment. In Y. Rosen, S. Ferrara, & M. Mosharraf (Eds.), *Handbook of research on technology tools for real-world skill development* (pp. 750–777). IGI Global.

Junker, B. W., Patz, R. J., & VanHoudnos, N. M. (2016). Markov chain Monte Carlo for item response models. In W. J. van der Linden (Ed.), *Handbook of item response theory, volume two: statistical tools* (pp. 271–325). CRC Press.

Kerr, D., Chung, G., & Iseli, M. (2011). *The feasibility of using cluster analysis to examine log data from educational video games* (CRESST Report No. 790). University of California, National Center for Research on Evaluation, Standards, and Student Testing (CRESST), Center for Studies in Education, UCLA. https://files.eric.ed.gov/fulltext/ED520531.pdf

Kim, J. S., & Bolt, D. M. (2007). Estimating item response theory models using Markov chain Monte Carlo methods. *Educational Measurement: Issues and Practice, 26*(4), 38–51.

König, C., Spoden, C., & Frey, A. (2020). An Optimized Bayesian Hierarchical Two-Parameter Logistic Model for Small Sample Item Calibration. *Applied Psychological Measurement, 44*(4), 311–326. http://doi.org/https://doi.org/10.1177/0146621619893786

LaMar, M. M. (2018). Markov decision process measurement model. *Psychometrika, 83*(1), 67–88. https://doi.org/10.1007/s11336-017-9570-0

Levy, R., & Mislevy, R. J. (2016). *Bayesian psychometric modeling*. CRC Press.

Liao, D., He, Q., & Jiao, H. (2019). Mapping background variables with sequential patterns in problem-solving environments: An investigation of US Adults' employment status in PIAAC. *Frontiers in Psychology, 10*, 646. https://doi.org/10.3389/fpsyg.2019.00646

Liu, H., Liu, Y., & Li, M. (2018). Analysis of process data of PISA 2012 computer-based problem solving: Application of the modified multilevel mixture IRT model. *Frontiers in Psychology, 9*, Article 1372. https://doi.org/10.3389/fpsyg.2018.01372

OECD. (2014). *PISA 2012 results: Creative problem solving: Students' skills in tackling real-life problems (Volume V)*. OECD Publishing. http://www.oecd.org/education/pisa-2012-results-volumev.htm

Patz, R. J., & Junker, B. W. (1999a). Applications and extensions of MCMC in IRT: Multiple item types, missing data, and rated responses. *Journal of Educational Behavioral Statistics, 24*(4), 342–366. https://doi.org/10.3102/10769986024004342

Patz, R. J., & Junker, B. W. (1999b). A straightforward approach to Markov chain Monte Carlo methods for item response models. *Journal of Educational Behavioral Statistics, 24*(2), 146–178. https://doi.org/10.3102/10769986024002146

Qiao, X., & Jiao, H. (2018). Data mining techniques in analyzing process data: a didactic. *Frontiers in psychology, 9*, 2231.

R Core Team. (2018). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing.

Rosenthal, J. S. (2011). Optimal proposal distributions and adaptive MCMC. In S. Brooks, A. Gelman, G. L. Jones, & X.-L. Meng (Eds.), *Handbook of Markov Chain Monte Carlo* (pp. 93–111). Chapman and Hall/CRC.

Sclove, L. S. (1987). Application of model-selection criteria to some problems in multivariate analysis. *Psychometrika, 52*, 333–343.

Shu, Z., Bergner, Y., Zhu, M., Hao, J., & von Davier, A. A. (2017). An item response theory analysis of problem-solving processes in scenario-based tasks. *Psychological Test and Assessment Modeling, 59*(1), 109–131.

Spiegelhalter, D. J., Best, N. G., Carlin, B. P., & Van der Linde, A. (1998). *Bayesian deviance, the effective number of parameters, and the comparison of arbitrarily complex models*. Technical report, MRC Biostatistics Unit.

Stadler, M., Hofer, S., & Greiff, S. (2020). First among equals: Log data indicates ability differences despite equal scores. *Computers in Human Behavior, 111*, 106442. https://doi.org/10.1016/j.chb.2020.106442

Schwarz, G. (1978). Estimating the dimension of a model. *Annals of Statistics, 6*, 461–464.

Tang, X., Wang, Z., He, Q., Liu, J., & Ying, Z. (2020). Latent Feature Extraction for Process Data via Multidimensional Scaling. *Psychometrika, 85*(2), 378–397.

Tang, X., Wang, Z., Liu, J., & Ying, Z. (2021). An exploratory analysis of the latent structure of process data via action sequence autoencoders. *British Journal of Mathematical and Statistical Psychology, 74*(1), 1–33. https://doi.org/10.1111/bmsp.12203

Xiao, Y., He, Q., Veldkamp, B., & Liu, H. (2021). Exploring latent states of problem-solving competence using hidden Markov model on process data. *Journal of Computer Assisted Learning*. Advanced Online. *Publication.* https://doi.org/10.1111/emip.12474

Zhan, P., & Qiao, X. (2022). Diagnostic Classification Analysis of Problem-Solving Competency Using Process Data: An Item Expansion Method. *Psychometrika. Advanced Online Publication.* https://doi.org/10.1007/s11336-022-09855-9

Open Practice Statement The data in the empirical study reported here is available at OECD website: http://www.oecd.org/pisa/pisaproducts/database-cbapisa2012.htm.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.