Real-time prediction and adaptive adjustment of continuous casting based on deep learning

Ziqing Lu[1], Neng Ren[1], Xiaowei Xu[1], Jun Li[1,*], Chinnapat Panwisawas[2], Mingxu Xia[1], Hongbiao Dong[3], Eric Tsang[4], Jianguo Li[1]

[1]Shanghai Key Laboratory of Advanced High-temperature Materials and Precision Forming, School of Material Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China.

[2]School of Engineering and Materials Science, Queen Mary University of London, London E1 4NS, United Kingdom.

[3]School of Engineering, University of Leicester, Leicester LE1 7RH, United Kingdom

[4]School of Computer Science and Engineering, Macau University of Science and Technology, Taipa, Macau, China

Corresponding author: Jun Li (li.jun@sjtu.edu.cn, https://orcid.org/0000-0003-4915-0336)

Abstract

Digitalisation of metallurgical manufacturing, especially technological continuous casting using numerical models of heat and mass transfer and subsequent solidification has been developed to achieve high manufacturing efficiency with minimum defects and hence low scrappage. It is still challenging to perform adaptive closed-loop process adjustment using high-fidelity computation in real-time. To address this challenge, surrogate models are a good option to replace the high-fidelity model, with acceptable accuracy and less computational time and cost. Based on deep learning technology, here we developed a real-time prediction (ReP) model to predict the three-dimensional (3D) temperature field distribution in continuous casting on millisecond timescale, with mean absolute error (MAE) of 4.19 K and mean absolute percent error (MAPE) of 0.49% on test data. Moreover, by combining the ReP model with machine learning technology—Bayesian optimization, we realised the rapid decision-making intelligent adaptation of the operating parameters for continuous casting with high predictive capability. This innovative and reliable method has a great potential in the intelligent control of the metallurgical manufacturing process.

Introduction

Continuous casting (CC) is a sophisticated metallurgical process used to manufacture most of the technological steel products (including billets, blooms, and slabs) around the world. There are always several types of casting defects including surface and corner cracks, center macrosegregation, center shrinkage, and porosity in the CC billets, which reduced the working performance of the steel products. An effective solution to overcome these defects is to use soft reduction technology, in which the key technical parameters are the position and amount of reduction, and appropriate secondary cooling water control. For the effective control of the secondary cooling and the soft reduction, the thickness of the solidified shell and the metallurgical length (the distance where solidification is sufficiently complete in slab center) in the CC process need to be accurately estimated[1]. Therefore, it is of great importance to analyse altogether the heat transfer, solidification, multiphase turbulent flow, and other interacting phenomena in the CC process. However, it is rather difficult to perform enough experiments on CC process to generate data for digitalization of the process, and due to the limitation of measurement techniques and harsh production environment, most of the important information in CC processes cannot be obtained by direct experiment. Numerical simulation is widely applicable to gain a better understanding on these fundamental behaviors, and subsequently predict the 3D temperature distribution and solidification shell, which are key information for process optimisation and defect mitigation strategy.

Direct numerical simulation (DNS) model based upon computational fluid dynamics (CFD) calculation for CC has greatly improved since the 1980s[2], and has been used to optimise the operating

conditions, such as the cooling water arrangement in the secondary cooling zone, resulting in the reduction of casting defects in the strand. The DNS models have been developed to simulate the complex phenomena during the CC process[3-6], and studies have combined the DNS with optimisation-regulation algorithms[7-10], in order to speed up the optimisation process for a more appropriate arrangement of cooling water. However, a common problem of these models is the excessive computational time and resource consumption, especially in parametric studies for the process optimisations where a great number of DNS calculations need to be performed, even in serial. Besides, there are always situations where the actual process behaviours misbehave an unexpected way. In this case, a fast decision and an autocorrect response are intensely needed to prevent the process from becoming more exacerbated.

Deep learning (DL) has emerged as a powerful technology, exhibiting state-of-the-art performance on a variety of tasks. With its exceptional ability to learn from vast amounts of data, DL techniques has been widely used in CC process[11] to achieve continuous monitoring (3D laser image scanning system based on binocular imaging and DL techniques to detect, recognise, classify, and delineate the defects in CC product surfaces[12].), control (temperature control optimisation[13] and molten steel temperature preset[14] in the CC process with deep neural networks.), and assessment of the implementation (internal crack prediction[15] and breakout prediction[16] in the CC process with deep neural networks.). Despite the benefits that DL techniques have offered for improving the intelligence and efficiency of the CC process, there is still a pressing need to develop methods that enable fast decision-making and rapid autocorrection response in this field.

For real-time monitoring and fast response of heat transfer and solidification phenomena in the

CC process, by simplifying the numerical model[17,18] and enhancing computing resources[19-21] could help speed up the process optimisation; still these approaches have to compromise the accuracy and/or require unacceptable computing resources. Whereas the conventional DNS technology cannot offer real-time digital representation considering the demanding computational time, resources and accuracy. To circumvent the limitations of conventional DNS techniques, researchers have attempted to develop surrogate models using DL techniques to predict the DNS results with acceptable accuracy and less computational time and cost. For instance, CNNs-based autoencoder has been utilised to predict CFD velocity field by signed distance function[22,23]. To deal with irregular geometry grid, fully connected neural networks has been employed to predict the temperature evolutions calculated by finite element models[24]. In order to establish a mapping from the parametric space of the problem to its solution space, Nikolopoulos et al. applied a CNNs-based autoencoder and a feed-forward neural network to efficiently map points from the parametric space to the compressed version of the respective solution matrices[25]. Since the DNS results are highly related to the corresponding technological parameter setting, the DL model can be used to learn the relationships between them, so as to achieve rapid prediction of DNS results under the corresponding parameter settings. However, for different DNS models and varied prediction needs, building the corresponding dataset is challenging and resource-intensive, and more efforts are required to select and establish the appropriate DL models to build the corresponding surrogate models. Although DL-based surrogate model for CC has been reported[26], which incorporated CNNs and recurrent neural networks to address both spatial and sequential information, it is limited to a 2D simulation prediction and focuses more on time series prediction.

In this study, we established an efficient surrogate model for a 3D CC heat transfer DNS process

and demonstrated its potential industrial application. The proposed real-time prediction (ReP) model is capable of computing the 3D temperature fields of CC process with the aid of DL techniques, and combining with Bayesian optimisation (BO) to conduct the intelligent adaptation, it can improve the key operating conditions to achieve the expected target. The difference and contribution between our study and prior work are listed as followed: 1). Different tasks: Our surrogate model aims to predict the 3D temperature field during CC process under different casting speed, cooling water flow rate. 2). Different approaches: For our specific data form and task requirements, we have designed a specific model structure to achieve the best performance. 3). Exploration of application prospect of surrogate model: We aim to demonstrate how the surrogate model can be applied to process decision making and, in an attempt, to serve real-world metallurgical manufacturing engineering to meet the needs of intelligent control. We first developed a combined hybrid 3D/2D model[3] to produce the CC DNS data. With the established 800 dataset of 3D simulated temperature fields (700 for training and 100 for testing), we trained a convolutional neural network (CNN)-based autoencoder[27] to extract the latent code from the data, and reconstructed the data using the latent code. Then we developed an MLP-Mixer-based[28] parameter encoder to map the technological parameter setting (casting speed and cooling water flow rate in eight different cooling zone) to the corresponding latent code. The ReP model can rapidly predict the 3D temperature fields accurately on millisecond timescale, without the requirement for excessive resources.

Results

Model construction

We hypothesise that the CC DNS temperature field (T_f) is a complex function F of the corresponding technological parameter setting (p), as shown Eq. 1.

$$T_f = F(p) \quad (1)$$

A real-time prediction (ReP) model has been successfully developed to fit function F to predict the temperature field under the corresponding technological parameter setting. The process is divided into two parts: encoder-decoder structure autoencoder and parameter encoder, as shown in Fig. 1. To extract the most valuable featured information from the CC data, we trained a self-supervision CNN-based autoencoder with the DNS data as the input and the output; the autoencoder needs to compress the 3D DNS data to a one-dimensional 128×1 latent code, and reconstruct the DNS data, so the autoencoder can extract the essential information from the data by the encoder part, and reconstruct the data by the decoder part using Eq. 2 and Eq. 3, where l_c, T_f, T_f', E, D represent latent code, DNS temperature field, reconstructed temperature filed, encoder and decoder, respectively.

$$l_c = E(T_f) \quad (2)$$

$$T_f' = D(l_c) \quad (3)$$

The Multilayer Perception (MLP) layer placed at the end of the encoder and the beginning of the decoder works as a nonlinear projection head that produces the latent code, which can improve the representation quality[29]. Considering symmetry and computational cost, we only take a quarter of the 1000×78×26 3D domain (20362×1530×190 mm^3), then split it by layers, and zero pad it into thirteen

1000×40×1 data for training, as shown in Fig. 1.

With the well-trained autoencoder, then we trained a parameter encoder to map the technological parameter sets to the corresponding latent code. As our autoencoder extracts thirteen layers of 128×1 latent codes for each 3D data points, the autoencoder can only 'see' one layer of the 3D data at a time. As a result, the information between layers in the 3D temperature field is completely ignored. Since the temperature between two layers is interrelated, we consider this information in the parameter encoder part. Therefore, we employed an MLP-Mixer-based parameter encoder to map the 9×1 technological parameter setting to the corresponding latent code (128×13) of 3D data, as shown in Eq. 4, where l'_c, p, P are the latent code prediction, technological parameter setting and the parameter encoder, respectively. We proved the effectiveness in this approach by reintroducing the information between layers in the parameter encoder part (see Supplementary Figure 4).

$$l'_c = P(p) \tag{4}$$

The structures of autoencoder and parameter encoder are selected after experimental verification (see Supplementary Figure 3-5) to ensure the best performance. The loss function and evaluation metric are described with mean absolute error (MAE) as Eq. 5 and mean absolute percent error (MAPE) as Eq. 6, where $T_{i,x,y,z}$, $T'_{i,x,y,z}$ and n are ground truth values, predicted values and total sample number.

$$\mathrm{MAE}(T,T') = \frac{1}{n}\frac{1}{1000\times 39\times 13}\sum_{i=1}^{n}\sum_{x=1}^{1000}\sum_{y=1}^{39}\sum_{z=1}^{13}\left|T_{i,x,y,z}-T'_{i,x,y,z}\right| \tag{5}$$

$$\mathrm{MAPE}(T,T') = \frac{1}{n}\frac{1}{1000\times 39\times 13}\sum_{i=1}^{n}\sum_{x=1}^{1000}\sum_{y=1}^{39}\sum_{z=1}^{13}\left|\frac{T_{i,x,y,z}-T'_{i,x,y,z}}{T_{i,x,y,z}}\right| \tag{6}$$

Real-time prediction results

With the well-trained autoencoder and parameter encoder models, we can complete the 3D temperature field prediction, as shown in Fig. 2(a) and Eq. 7, where T_{f}'', p, D, P are the temperature field prediction, technological parameter setting, decoder and parameter encoder, respectively. The 9×1 technological parameter setting is first encoded into 128×13 latent code by the parameter encoder, then the decoder will decode the latent code into temperature field prediction.

$$T_{\mathrm{f}}'' = D\big(P(p)\big) \tag{7}$$

A typical 3D temperature field predicted by ReP model is shown in Fig. 2(a). In order to verify the overall reliability of the ReP model, we compared the 3D temperature fields predicted by ReP model and DNS respectively, as shown in Fig. 2(b). The predicted results by ReP model are in good agreement with the DNS results.

The MAPE result of test data can better reflect the performance of the model, since the test data is isolated from the training process. Therefore, to demonstrate the performance of the model, we mainly use the MAPE result of the test data for the verification. In Fig. 3(a), the MAPE distribution of the ReP model on 100 test data is plotted. First, we ensure the uniformity of data points by sampling algorithms for the accuracy and robustness of our model. The MAPE is very low on most of the test data (blue points, MAPE ≤ 0.5%: 68, 0.5 < MAPE ≤ 1.0%: 20). Though the error is a bit higher on a small group of test data (red point, 1.0% < MAPE ≤ 1.5%: 10, 1.5% < MAPE ≤ 2.1%: 2), these data is clustered at the origin and corresponding to the very low cooling water flow rate (CWFR), which is rarely used in industry.

To show the accuracy of the ReP model at each data point more deeply, we plotted the ReP temperature and DNS temperature results in Fig. 3(b). However, because the total number of the test data points is too large (100 test data, each contains 507000 (1000×39×13) data points), we uniformly selected 27 points from each test data (in each 1000×39×13 test data, selected points at 100, 500, 900 in the first dimension, 10, 20, 30 in the second dimension, 1, 5, 9 in the third dimension), 2700 data points in total. As shown in Fig. 3(b), the ReP temperature results fits the DNS temperature results very well, with R^2 score of 0.9982. And the R^2 score between the ReP and DNS results on the whole test data point is 0.9987.

Furthermore, we demonstrate the accuracy of the ReP model in two different dimensions: along casting direction (Fig. 3(c)) and on the transverse section (Fig. 3(d)). We calculated the average MAPE at a certain location or section over all the test data to analyse the accuracy of the model in depth. In Fig. 3(c), the solid red curve shows the average MAPE on all test data at different distances from the meniscus. The value of MAPE starts nearly zero at the beginning and then increases with the distance from the meniscus, reaching 0.91% at the bottom of the slab. We believe that this phenomenon is related to the complexity of the temperature fields in the data. For example, since in our case the temperature at the beginning of the slab (the mould) is affected only by the casting speed under the same pouring temperature, the ReP model can learn this relationship easily. As the distance from the meniscus increases, the temperature is influenced by more and more factors, the casting speed, the CWFR in the current cooling zone and the previous zone. It is therefore getting harder for the ReP model to make an accurate prediction. To demonstrate this, we calculated the standard deviation of the temperature at different distance in all the test data, as illustrated in the blue curve of Fig. 3(c). It can

be inferred that the non-uniformity distribution of the error is consistent with the standard deviation, which is confirmed by the similarity of the error and standard deviation distribution along the casting direction. Similarly, the average MAPE is calculated on all test data on the transverse section, as shown in Fig. 3(d). The MAPE on the surfaces is relatively larger, especially on the lateral surfaces where it reaches the maximum value of 1.78%; in the interior of the slab, it is very small, value of which ranges from 0.24% to 0.6%. Overall, by analysing the errors in two different dimensions, it can be concluded that the errors of the prediction by the ReP model are relatively small, this has confirmed further the validated reliability of the model.

The ReP model can effectively make real-time prediction on the 3D temperature field of the continuous casting process within only 0.12 s (on personal laptop with CPU: AMD Ryzen 7 5800H and GPU: NVIDIA GeForce RTX 3060, see Supplementary Movies 1) and high accuracy (on test data, MAE: 4.19 K, MAPE: 0.49%; on training data, MAE: 4.19 K, MAPE: 0.48%; the standard deviation of the total data points (800×1000×39×13) is 181.7 K). In contrast, the conventional DNS model costs about 8 hours with 4 CPUs (Intel Xeon E5-2620, 2.40 GHz, 32 G RAM for each CPU) in parallel (Intel MPI) to complete the DNS process.

Adaptive adjustment of the secondary cooling

The secondary cooling is an important factor affecting metallurgical length (the distance where solidification is sufficiently complete in slab center, calculated based upon solid fraction along the slab direction as shown in Fig. 3(a)), which is a crucial processing variable used to estimate the casting

defects distributed along the centerline of the slab, such as centerline segregation, porosity, inclusions, alloy-rich regions, and even cracks. They are especially harmful in rolling process of the highly alloyed steel slabs[30,31]. Robust and accurate control of secondary cooling is vital to prevent or even suppress the defects and to the produce high-quality steel slabs. Real-time control of secondary cooling to control the metallurgical length is highly desirable to meet the demands of product quality and operational safety. Thus far we can predict the temperature field on millisecond timescale, the ReP model can provide the information for the adaptive real-time closed loop process control. To narrow down the enormous search space during adaptive adjustment, we further combine the quick prediction model with Bayesian optimisation (BO)[32] to solve when one or more technological parameters (casting speed and cooling water flow rate in eight different cooling zone) change, and interrogate how to set other parameters to keep the metallurgical length.

We randomly choose a predicted result as the initial state, and increase the casting speed from 1.3 $m \cdot min^{-1}$ to 1.37 $m \cdot min^{-1}$. The metallurgical length is lengthened from 9.36 m to 10.18 m. Then, we use the BO to search for the best CWFR settings for Zone 1 and Zone 2 to minimise the change of the metallurgical length. The BO will provide a new prediction parameter setting based on Gaussian process. Then the new parameter setting is input to our ReP model to predict the 3D temperature field. The next step is to calculate the objective function and iterate the Bayesian model as shown in Fig. 4(a). The original, mutation and new parameter setting are listed in Table 1. Here, we use the difference of the metallurgical length as the objective function, and the solidification state is calculated according to the temperature field by Eq. 8, where f_L, T, $T_{Solidus}$ and $T_{Liquidus}$ are liquid fraction, temperature (K), solidus temperature (1715 K), and liquidus temperature (1786 K), respectively. In this experiment, it

takes BO 22 iterations to converge, and a new parameter setting leading to the same metallurgical length is found. The comparison of the shell thickness under the original, mutation and new parameter settings is shown in Fig. 4(b). The thickness gets thinner with the increase in casting speed, and besides the BO finds a higher CWFR in Zone 1 and Zone 2 arrangements to get a 9.32 m metallurgical length, which is similar to the original one. A metallurgical length difference map is shown in Fig. 4(c), and some points during the BO iterative process are drawn to show the searching path. The map shows that the lowest difference locates in an arc range. This corresponds to the fact that the cooling water flow rate in Zone 1 and Zone 2 should be complementary, and either too low or too high CWFR will result in a larger offset. With the help of our ReP model, this optimisation process only takes 5.2 s, while it would be days for the conventional DNS.

$$f_{\mathrm{L}} = \begin{cases} 0 & T \le T_{\mathrm{Solidus}} \\ \dfrac{T - T_{\mathrm{Solidus}}}{T_{\mathrm{Liquidus}} - T_{\mathrm{Solidus}}} & T_{\mathrm{Solidus}} < T < T_{\mathrm{Liquidus}} \\ 1 & T \ge T_{\mathrm{Liquidus}} \end{cases} \tag{8}$$

The experiment above uses a target metallurgical length as the objective function. Moreover, other objective functions, such as target temperature distribution, target shell thickness, and target temperature at a certain position, can be set for technological parameter searching to achieve different quality objectives and even a hybrid one. In addition, other optimisation regulation algorithms, such as simulated annealing, differential evolution, and particle swarm optimisation, can also be combined with our model. Only a few seconds are needed for the ReP model to run hundreds of iterations for the optimisation regulation algorithms and to ensure the convergence. So it is suitable to deal with the complex changes and requirements in real manufacturing environment. Besides, our well-trained ReP model is more deployment friendly and can be performed on a regular personal computer. As shown

253 in Fig. 5, the training of the ReP model is a one-time cost, the low computing time and cost makes the
254 adaptive adjustment system have the advantages of edge deployment and control, and realized
255 intelligence casting to improve the steel quality and reduce costs.
256

257 **Dataset volume analysis**

258 As a data-driven model, the performance and robustness of DL model are very sensitive to the
259 datasets size, and small size datasets impose great challenge in developing DL model. In this study, we
260 simulated 700 CC DNS temperature field data to train our DL model and achieved satisfactory
261 performance. In order to investigate the impact of dataset size on model performance and explore ways
262 to improve, we first illustrate the effect of data volume on model performance by analysing the
263 relationships between training data and test data, and then compare the test results under different
264 training data volumes.

265 In Fig. 6(a), we show the MAPE distribution of 100 test data, where the cosine similarity between
266 the test technological parameter setting and the whole training technological parameter setting is set
267 as the x-axis. Higher cosine similarity means that this test data is closer to the coverage of the training
268 data, which should result in better performance on this test data. For a better view, we divide the region
269 in Fig. 6(a) into seven sub-regions equally and draw box-whisker plots for the last five sub-regions in
270 Fig. 6(b), we can see the trend of MAPE decreasing with the increase in cosine similarity. Besides, we
271 compare the MAPE results when we reduce the training data volume from 700 to 200, as shown in Fig.
272 6(c). Obviously, the MAPE results deteriorate as the training data volume reduces. Due to the too many

(nine) technological parameters and the too large sampling space, our training data cannot cover the entire sampling space, so it is hard to map the relationships between the technological parameter sets and the latent code perfectly. Increasing the data volume to cover more sampling space should improve the performance. But we find a diminishing marginal effect when changing dataset volume, as shown in Fig. 6(c). Therefore, the amount of data to achieve a perfect result might be rather extensive. When the amount of data is limited, the performance and robustness of the model can be improved through data augmentation[33], regularization[34], transfer learning[35], and knowledge distillation[36].

However, when comparing the results of reducing the training data volume of the autoencoder and reducing the training data volume of the parameter encoder (when reducing the training data volume of one model, the training data volume for another model stays at 700). The MAPE curve is similar between the red circle curve (reducing the training data volume of both the autoencoder and parameter encoder) and the blue triangle curve (reducing the training data volume of the parameter encoder), but the MAPE results do not deteriorate much when only reducing the training data volume of the autoencoder (yellow square curve). It can be inferred that the bottleneck of our ReP model is the parameter encoder, and the autoencoder can be well-trained with only a small amount of data. The loss curve also shows that the parameter encoder is a little bit underfitting while the autoencoder fits well (see Supplementary Figure 7-8). Fig. 6(d) shows the comparison of ReP results by different training data volumes. Clearly, the ReP result is improved with the increase in the number of training data.

Discussion

In this study, we developed a ReP model to predict the 3D CC temperature field with high precision and throughput. Proving that the DL technology can learn the relationships between the DNS data and the corresponding technological parameter setting, and then make accurate predictions. The final performance and accuracy of the Rep model depends on two aspects: accuracy of the numerical simulation and completeness of dataset space; and the accuracy of DL model. Therefore, to make more precise prediction, a better solidification model is essential. As a first attempt for this approach, the main purpose of this study is to verify the feasibility of the approach, so we do not give much consideration to the CC DNS dataset, and current dataset contains only the temperature information of the CC. In fact, when we use the most advanced DNS models (such as macrosegregation model and dendritic structure model) to generate datasets, these CC DNS datasets will contain more valuable information (process variables), and we can fully develop ReP models for macrosegregation and dendritic structure prediction, which is of great significance for the quality and intelligent control of CC process, but lots of efforts and times may be required to build this kind of dataset. In addition, this approach is content independent in some ways, so it is possible to be spread to other research areas to actualise similar ReP processes to help accelerate scientific research.

Furthermore, we take the advantages of the ReP model and combine it with optimisation regulation algorithms to achieve fast adaptive adjustment. In our model, the adaptive adjustment is completed in seconds, which is much faster than using conventional DNS technology, demonstrating the potential application scenarios and capabilities. The trained ReP model does not require extensive

computing resources as the conventional DNS, which is more computational friendly for edge deployment and computing. Thus, it is also possible to deploy the ReP model at the front of the fabrication to help implementation fast adaptive adjustment to improve process control as shown in Fig. 5.

Besides, it is necessary to discuss the limitations and shortcomings of our model as well:

- As a data-driven method, the dataset is a common concern for researchers. Although a lot of effort has been made to produce a large number of the CC dataset, our ReP model has yet reached a perfect accuracy. As illustrated in Fig. 6(c), the dataset, especially for parameter encoder, needed to be further enriched to achieve higher accuracy of the trained ReP model. Moreover, this work provides the most complete dataset to digitalise the metallurgical process.
- The ReP model can only predict the results within the training data space. For example, current ReP model is difficult to accurately predict the CC results with a casting speed larger than 1.65 $m \cdot min^{-1}$, because the operating parameter is beyond the range of the operating conditions (0.75 ~ 1.65 $m \cdot min^{-1}$) in the training data. We used the ReP model to predict the temperature at the end of the slab under the extreme casting process conditions, i.e. the minimum casting speed and maximum cooling water flow, and obtained a minimum temperature of 598K, but the actual value should be 372 K from DNS result. The reason is that the minimum temperature in our dataset is 604 K, and the predicted minimum temperature will be limited by the training dataset and not much lower than the lowest temperature in this dataset. Therefore, to obtain higher prediction accuracy, it is necessary not only to increase the dataset, but also to expand the range of processing processes and resultant temperatures as much as possible.

More work is needed to completely replace the conventional DNS. Having said that our model has excellent advantages in terms of computational speed and resources, and it is accurate enough under the conditions investigated. The digitalisation tool proposed here is therefore valuable for accelerating the manufacturing science research and technology take-up.

Methods

Data preparation and preprocessing

In this study, we developed a combined hybrid 3D/2D model for heat transfer, fluid flow, and solidification simulation using computational fluid dynamics (CFD) calculation, according to the reference[3] to generate our dataset (see Supplementary Figure 1-2, Supplementary Table 1-3 and Supplementary Note 1). The solution of the CC model is performed on the ANSYS FLUENT 14.5 CFD software. We chose this model for three reasons. First, it is a verified model that can provide reliable results. Second, the 3D simulation results are more representative than the 2D simulation ones. Third, the computational speed is much faster than a 3D simulation model, which is a very important consideration because a great number of simulations need to be performed for big datasets. This numerical model divided the computational domain into two parts—the 3D turbulent flow region and the 2D laminar flow region. The velocity of the molten steel in the casting direction is equal to the casting speed after the 3D turbulent flow region and 2D laminar flow region interface (which is set at the end of Zone 3) and thus forming plug flow. Moreover, as reported[37], the heat flux in the casting

direction accounts for just a little (about 3%-6%) of the total heat loss. These phenomena enable us to ignore the heat flux in the casting direction in the laminar flow region. In addition, due to the heat transfer, cooling condition, and solidification process of slab continuous casting are of good symmetry in the width and thickness directions of the slab, the difference in cooling intensity between the inner and outer wide surfaces of the slab, and the effect of the bending and straightening process on the thermal contraction deformation, can be neglected to reduce the computational cost[38]. One-quarter of the strand was included in the computational domain. The computation cost has obviously reduced, providing us an advantage for generating a large amount of data. We employed this CC model to produce our dataset under different casting speeds and cooling water flow rates. In addition, this model considers the uneven distribution of cooling water in the wide face direction according to the actual process, and there are eight secondary cooling zones with different cooling water rates.

The DNS model produced the 20362×765×95 mm^3 temperature field (a quarter of the slab). In other words, a data dimension of 1000×39×13. However, this data shape was too large for a DL model. In order to reduce the computational cost, we divided the data into 13 layers in the thickness direction (Z-direction) as illustrated in Fig. 1. Moreover, we zero pad the data in the width direction (Y-direction) to shape 1000×40×1 and normalise them between 0 and 1 to simply the scaling operation. We used Latin hypercube sampling (LHS)[39] to collect our sampling points to guarantee the randomness and uniformity of the dataset. The sampling range for each parameter is listed in Supplementary Table 4. 3D DNS data of 800 model outputs was calculated, and we divided the dataset into 700 dataset for training and 100 dataset for testing. Then we expanded the training data through data augmentation[33]. The test data were only used to test the model performance, instead of participated in any training

processes.

Real-time prediction model

The framework of the whole ReP model is shown in Fig. 1, the backbone of our autoencoder and parameter encoder is the ConvNeXt block and Mixer layer, which are built according to the corresponding references[28,40] (see Supplementary Figure 6). The Down/Up-sample layer is a 2×2 Conv2D/Conv2DTranspose with strides 2 to half/double the height and width of the feature map. The projection layer is a 1×1 Conv2D layer to change the dimension of the feature map. MLP layer is placed at the end of the encoder and the beginning of the decoder as a projection head to operate on one-dimension data, and is very important for extracting the latent code. A detailed structure is listed in Supplementary Table 5-6. The reason for separating the whole ReP model into autoencoder and parameter encoder is to introduce data augmentation to enhance the robustness of the model (see Supplementary Figure 9). Adam optimiser[41] was used for both two DL models at learning rates of 0.0013 and 0.01, respectively. Early stopping[42] and learning rate decay[43] were adopted to speed up the training progress. We set a 0.9 learning rate decay for five epoch patience and found it beneficial for training. The activation function and loss function for both two DL models were gelu[44] and MAE. The batch size for the autoencoder and parameter encoder is 32 and 128, respectively.

All DL models were trained on NVIDIA HGX A100 40GB GPU using the Tensorflow[45] library, and it took about seven hours to train the ReP model (about seven hours for the autoencoder and ten minutes for the parameter encoder).

Bayesian optimisation

As a framework for global optimisation of expensive-to-evaluate black-box functions, BO has become popular due to its remarkable performance in hyperparameter tuning of machine learning algorithms recently. The goal of Bayesian optimisation is to build a distribution based on previous measurements, priori information, using a Gaussian random process. It has been invented and used for a long time[46] and constantly upgraded and evolved[32,47]. In this work, a regular version of BO is achieved using the bayes_opt[48] library in Python, an efficient implementation of the BO methodology for nonlinear optimisation, experimental design and hyperparameter tuning. The initial step of random exploration is set to 5. With expected improvement function as acquisition function and set an exploration ratio of 0.1.

Data availability

Representative research data are given in the figures (and supplementary data). The source data for Fig. 3a, Fig. 3b and Fig. 3c is provided as Supplementary Data 1. The unedited raw data generated by Fluent that makes up the key dataset is accessible in figshare (doi link: https://doi.org/10.6084/m9.figshare.22810319.v1). Other generated and/or analysed datasets that support the findings of this study are available from the corresponding author upon reasonable request.

Code availability

All custom code used in this work, including that used to train and test ReP models, and perform secondary cooling water optimisation with BO, can be obtained from the following publicly accessible

GitHub page: https://github.com/LarkMi/ReP.

References

1 Chen, Z., Bentsman, J., Thomas, B. G. & Matsui, A. Study of spray cooling control to maintain metallurgical length during speed drop in steel continuous casting. *Iron and Steel Technology* **14**, 92-103 (2017).

2 Mills, K. *et al.* Looking into continuous casting mould. *Ironmaking & Steelmaking* **41**, 242-249 (2014).

3 Long, M. J. *et al.* A Combined Hybrid 3-D/2-D Model for Flow and Solidification Prediction during Slab Continuous Casting. *Metals* **8**, 10, doi:10.3390/met8030182 (2018).

4 Meng, Y. & Thomas, B. G. Heat-transfer and solidification model of continuous slab casting: CON1D. *Metallurgical and materials transactions B* **34**, 685-705 (2003).

5 Maurya, A. & Jha, P. K. Mathematical modelling of solidification in a curved strand during continuous casting of steel. *Journal of The Institution of Engineers (India): Series C* **98**, 45-52 (2017).

6 Ji, C., Cai, Z., Wang, W., Zhu, M. & Sahai, Y. Effect of transverse distribution of secondary cooling water on corner cracks in wide thick slab continuous casting process. *Ironmaking & Steelmaking* **41**, 360-368 (2014).

7 Miriyala, S. S., Subramanian, V. R. & Mitra, K. TRANSFORM-ANN for online optimization of complex industrial processes: Casting process as case study. *European Journal of Operational Research* **264**, 294-309 (2018).

8 Brezina, M., Mauder, T., Klimes, L. & Stetina, J. Comparison of Optimization-Regulation Algorithms for Secondary Cooling in Continuous Steel Casting. *Metals* **11**, doi:10.3390/met11020237 (2021).

9 Wang, Z., Zhang, Y., Jiang, Y., Zhang, J. & Zhang, S. An Optimization Control Method for Heat Transfer Model during Slab Continuous Casting. *Journal of Physics: Conference Series* **1575**, 012208 (2020).

10 Santos, C., Spim, J. & Garcia, A. Mathematical modeling and optimization strategies (genetic algorithm and knowledge base) applied to the continuous casting of steel. *Engineering applications of artificial intelligence* **16**, 511-527 (2003).

11 Cemernek, D. *et al.* Machine learning in continuous casting of steel: A state-of-the-art survey. *Journal of Intelligent Manufacturing*, 1-19 (2021).

12 Zhao, L. *et al.* A deep-learning-based 3D defect quantitative inspection system in CC products surface. *Sensors* **20**, 980 (2020).

13 Song, G. W. *et al.* Temperature Control Optimization in a Steel-Making Continuous Casting Process Using a Multimodal Deep Learning Approach. *steel research international* **90**, 1900321 (2019).

14 Yang, J., Zhang, J., Guo, W., Gao, S. & Liu, Q. End-point temperature preset of molten steel in the final refining unit based on an integration of deep neural network and multi-process operation simulation. *ISIJ International* **61**, 2100-2110 (2021).

15 Zou, L. *et al.* Internal Crack Prediction of Continuous Casting Billet Based on Principal Component Analysis and Deep Neural Network. *Metals* **11**, 1976 (2021).

16 Ansari, M. O. *et al.* Productivity Enhancement by Prediction of Liquid Steel Breakout during Continuous Casting Process in Manufacturing of Steel Slabs in Steel Plant Using Artificial Neural Network with Backpropagation Algorithms. *Materials* **15**, 670 (2022).

17 Petrus, B., Zheng, K., Zhou, X., Thomas, B. G. & Bentsman, J. Real-time, model-based spray-cooling control system for steel continuous casting. *Metallurgical and materials transactions B* **42**, 87-103 (2011).
18 Hashimoto, Y., Matsui, A., Hayase, T. & Kano, M. Real-time estimation of molten steel flow in continuous casting mold. *Metallurgical and Materials Transactions B* **51**, 581-588 (2020).
19 Klimes, L. & Stetina, J. Parallel dynamic solidification model of continuous steel casting on GPU. *Proceedings of 22nd Conference on metallurgy and materials, Ostrava*, 34-39 (2013).
20 Liu, X.-Y., Xie, Z., Yang, J., Meng, H.-J. & Wu, Z.-Y. A faster than real-time heat transfer model for continuous steel casting. *Journal of Materials Research and Technology* **19**, 4220-4232 (2022).
21 Wang, J. J., Meng, H. J., Yang, J. & Xie, Z. A fast method based on GPU for solidification structure simulation of continuous casting billets. *Journal of Computational Science* **48**, 101265 (2021).
22 Guo, X., Li, W. & Iorio, F. Convolutional neural networks for steady flow approximation. *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 481-490 (2016).
23 Ribeiro, M. D., Rehman, A., Ahmed, S. & Dengel, A. DeepCFD: Efficient steady-state laminar flow approximation with deep convolutional neural networks. *arXiv preprint arXiv:2004.08826* (2020).
24 Pham, T. Q. D. *et al.* Fast and accurate prediction of temperature evolutions in additive manufacturing process using deep learning. *Journal of Intelligent Manufacturing*, 1-19 (2022).
25 Nikolopoulos, S., Kalogeris, I. & Papadopoulos, V. Non-intrusive surrogate modeling for parametrized time-dependent partial differential equations using convolutional autoencoders. *Engineering Applications of Artificial Intelligence* **109**, 104652 (2022).
26 Lee, S. Y. *et al.* Spatial and sequential deep learning approach for predicting temperature distribution in a steel-making continuous casting process. *IEEE Access* **8**, 21953-21965 (2020).
27 Baldi, P. Autoencoders, unsupervised learning, and deep architectures. *Proceedings of ICML workshop on unsupervised and transfer learning*, 37-49 (2012).
28 Tolstikhin, I. O. *et al.* Mlp-mixer: An all-mlp architecture for vision. *Advances in Neural Information Processing Systems* **34**, 24261-24272 (2021).
29 Chen, T., Kornblith, S., Norouzi, M. & Hinton, G. A simple framework for contrastive learning of visual representations. *International conference on machine learning*, 1597-1607 (2020).
30 Sirgo, J. A., Campo, R., Lopez, A., Diaz, A. & Sancho, L. Measurement of centerline segregation in steel slabs. *Conference Record of the 2006 IEEE Industry Applications Conference Forty-First IAS Annual Meeting* **1**, 516-520 (2006).
31 Li, J., Xu, X.-w., Ren, N., Xia, M.-x. & Li, J.-g. A review on prediction of casting defects in steel ingots: from macrosegregation to multi-defect model. *Journal of Iron and Steel Research International*, 1-14 (2022).
32 Brochu, E., Cora, V. M. & De Freitas, N. A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *arXiv preprint arXiv:1012.2599* (2010).
33 Shorten, C. & Khoshgoftaar, T. M. A survey on image data augmentation for deep learning. *Journal of big data* **6**, 1-48 (2019).
34 Kukačka, J., Golkov, V. & Cremers, D. Regularization for deep learning: A taxonomy. *arXiv preprint arXiv:1710.10686* (2017).
35 Tan, C. *et al.* A survey on deep transfer learning. *Artificial Neural Networks and Machine Learning–ICANN 2018: 27th International Conference on Artificial Neural Networks, Rhodes, Greece, October 4-7, 2018, Proceedings,*

Part III 27, 270-279 (2018).

36 Gou, J., Yu, B., Maybank, S. J. & Tao, D. Knowledge distillation: A survey. *International Journal of Computer Vision* **129**, 1789-1819 (2021).

37 Brimacombe, J., Samarasekera, I. & Mahapatra, R. Basic knowledge and the achievement of quality in continuous casting. *IISC. The Sixth International Iron and Steel Congress.* **3**, 246-255 (1990).

38 Wu, C., Ji, C. & Zhu, M. Analysis of the Thermal Contraction of Wide-Thick Continuously Cast Slab and the Weighted Average Method to Design a Roll Gap. *steel research international* **88**, 1600514 (2017).

39 Damblin, G., Couplet, M. & Iooss, B. Numerical studies of space-filling designs: optimization of Latin Hypercube Samples and subprojection properties. *Journal of Simulation* **7**, 276-289, doi:10.1057/jos.2013.16 (2017).

40 Liu, Z. *et al.* A ConvNet for the 2020s. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11976-11986 (2022).

41 Kingma, D. P. & Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

42 Prechelt, L. in *Neural Networks: Tricks of the trade* 55-69 (Springer, 1998).

43 Loshchilov, I. & Hutter, F. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101* (2017).

44 Hendrycks, D. & Gimpel, K. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415* (2016).

45 Abadi, M. *et al.* Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467* (2016).

46 Zhilinskas, A. Single-step Bayesian search method for an extremum of functions of a single variable. *Cybernetics* **11**, 160-166 (1975).

47 Astudillo, R. & Frazier, P. Bayesian optimization of function networks. *Advances in Neural Information Processing Systems* **34**, 14463-14475 (2021).

48 Nogueira, F. Bayesian Optimization: Open source constrained global optimization tool for Python. *URL https://github.com/fmfn/BayesianOptimization* (2014).

Acknowledgements

This work was sponsored by National Natural Science Foundation of China (No. 52074182), Natural Science Foundation of Shanghai (No. 22ZR1430700) and the Guided Local Science and Technology Development Projects from the Central Government (No. 2021FRD05006). Chinnapat Panwisawas would like to acknowledge the support from UKRI Innovation Fellowship funded by EPSRC (EP/S000828/2). The computations in this paper were run on the Siyuan-1 cluster supported by the Center for High Performance Computing at Shanghai Jiao Tong University.

Author information

Authors and Affiliations:

Shanghai Key Laboratory of Advanced High-temperature Materials and Precision Forming, School of Material Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China.

Ziqing Lu, Neng Ren, Xiaowei Xu, Jun Li, Mingxu Xia, Jianguo Li.

School of Engineering and Materials Science, Queen Mary University of London, London E1 4NS, United Kingdom.

Chinnapat Panwisawas

School of Engineering, University of Leicester, Leicester LE1 7RH, United Kingdom

Hongbiao Dong

School of Computer Science and Engineering, Macau University of Science and Technology, Taipa, Macau, China

Eric Tsang

Authors contribution:

Ziqing Lu and Jun Li conceived the idea. Neng Ren and Xiaowei Xu built the continuous casting

numerical simulation model. Ziqing Lu carried out the calculations and analysis, and wrote the manuscript. Jun Li, Mingxu Xia and Chinnapat Panwisawas supervised the work. Hongbiao Dong, Eric Tsang and Jianguo Li contributed to the discussions and revisions of the manuscript.

Corresponding author

Correspondence to Jun Li.

Competing Interest

The authors declare no competing interests.

Additional information

Figure Legends

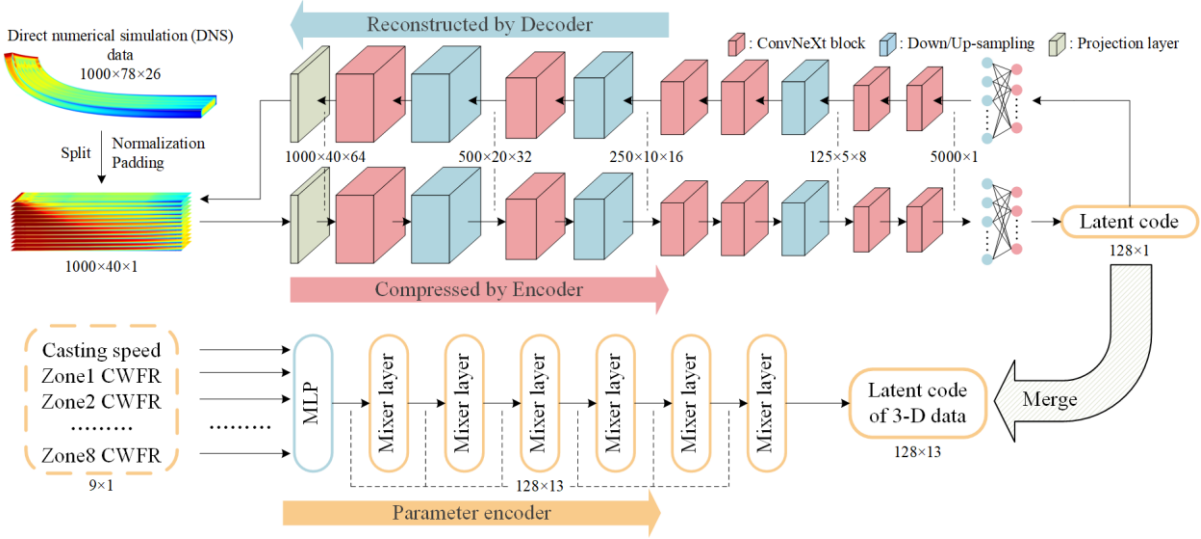

Fig. 1 Framework of the real-time prediction model. The 1000×78×26 simulated temperature field from computational fluid dynamic (CFD) calculation was split by layers, and then normalised and padded to thirteen 1000×40×1 shaped data to train the encoder-decoder structure autoencoder. The latent coder extracted by the encoder

part was merged to a 128×13 latent code of 3D dataset, to train the parameter encoder to map the technological parameter setting (casting speed and cooling water flow rate (CWFR) in each cooling zone) onto the corresponding latent code.

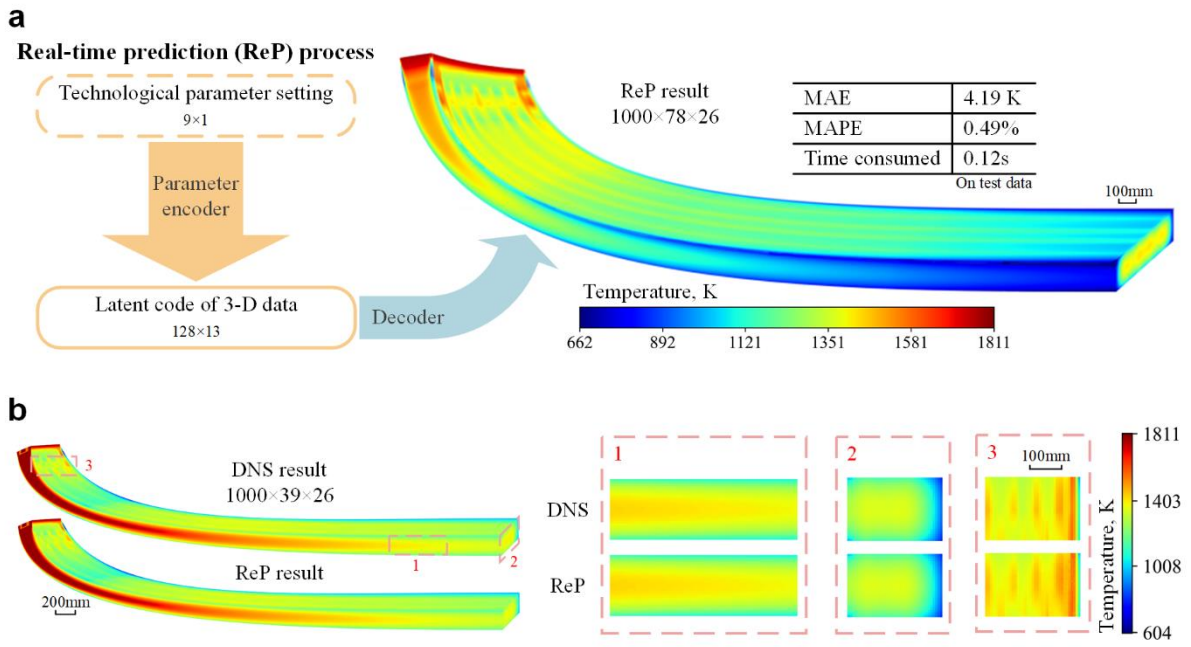

Fig. 2 Workflow and results of the real-time prediction (ReP) model **a** Workflow of the real-time prediction model. The parameter encoder map the 9×1 technological parameter set to the corresponding 128×13 latent code of 3D data, then the decoder part of the autoencoder predicts the 1000×78×26 3D temperature field by the latent code. It takes only 0.12 s for the real-time prediction model to complete. The mean absolute error (MAE) and mean absolute percent error (MAPE) on the test data are 4.19 K and 0.49%, respectively. **b** Comparison between 3D temperature fields obtained by ReP Model and direct numerical simulation (DNS). Half of the slab is shown, and three sub-regions are highlighted.

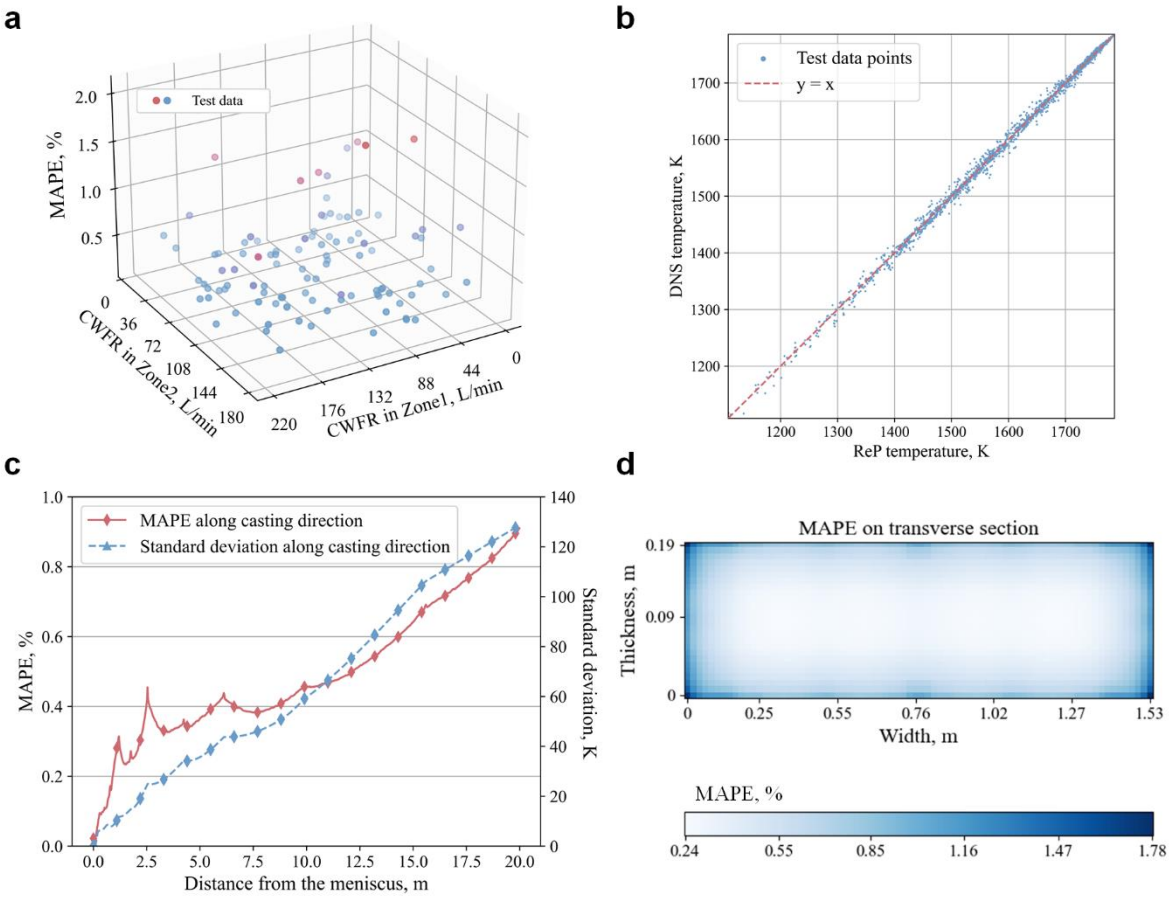

Fig. 3 Accuracy analysis of ReP model. **a** Mean absolute percent error (MAPE) distribution of 100 test data points, with cooling water flow rate (CWFR) in cooling zone 1 and cooling zone 2 as the x-axis and y-axis. Blue points represent MAPE values less than 1.0% and include 88 data points, with 68 data points have MAPE values less than 0.5% and 20 data points have MAPE values between 0.5% and 1.0%. Red points indicate MAPE values between 1.0% and 2.1%, with 10 data points having MAPE values between 1.0% and 1.5% and 2 data points having MAPE values between 1.5% and 2.1%. **b** Plot of ReP temperature and DNS temperature of selected test data points. Blue points represent the predicated temperature, red dashed line corresponds to the predicted value being the same as the DNS temperature. **c** MAPE and standard deviation distribution along casting direction in test data. Red curve shows the average MAPE on all test data at different distances from the meniscus, and blue dashed curve represent the standard deviation of the temperature at different distance in all the test data,. **d** MAPE distribution on the transverse section

in test data.

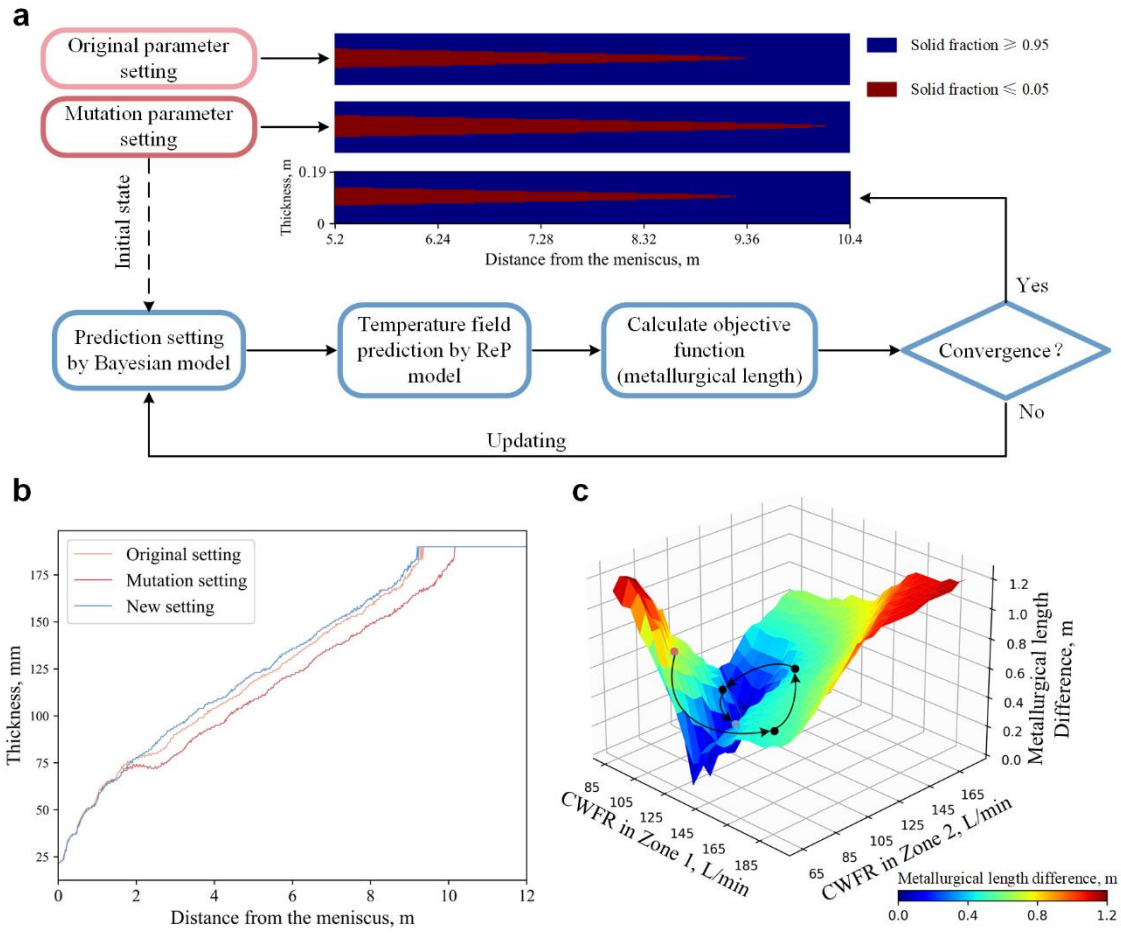

Fig. 4 Schematic of secondary cooling water optimisation using ReP model and Bayesian optimisation. **a** Workflow of the secondary cooling water optimisation. When the parameter setting changes, the mutation state is input to the Bayesian model as the initial state to obtain the optimised setting; then the new setting is used in the ReP model to predict the temperature field; finally, calculate the objective function to determinate whether to stop the optimisation search or not. **b** Comparison of the shell thickness curve under original setting (Orange curve), mutation setting (Red curve) and optimised setting (Blue curve). **c** Metallurgical length difference under the corresponding parameters (cooling water flow rate (CWFR)) mapped with ReP model. Points are scattered to show the search path of the BO process (Red points correspond to the start point, black point correspond to the intermediate point and blue point correspond to the end point).

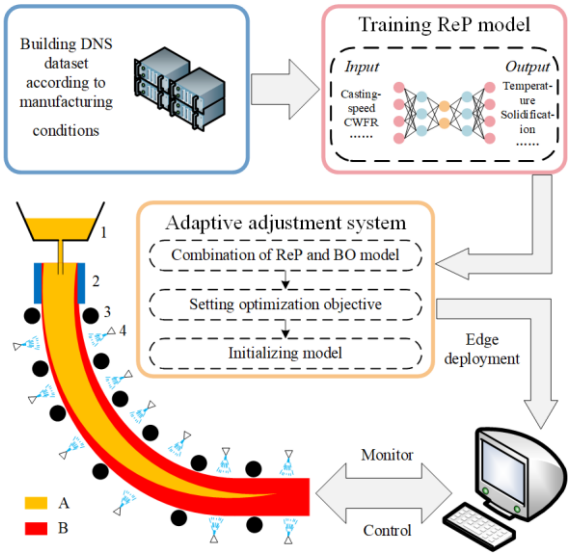

Fig. 5 Schematic of intelligence casting. First, we need to build DNS dataset to train the ReP model. Then, the ReP model can be combined with BO model to form the adaptive adjustment system, after setting the optimization objective and initializing the model, the system can be deployed at the front of the manufacturing factory to control the CC process. 1: Tundish. 2: Mold. 3: Roll support. 4: Spray nozzle. A: Liquid metal. B: Solidified metal.

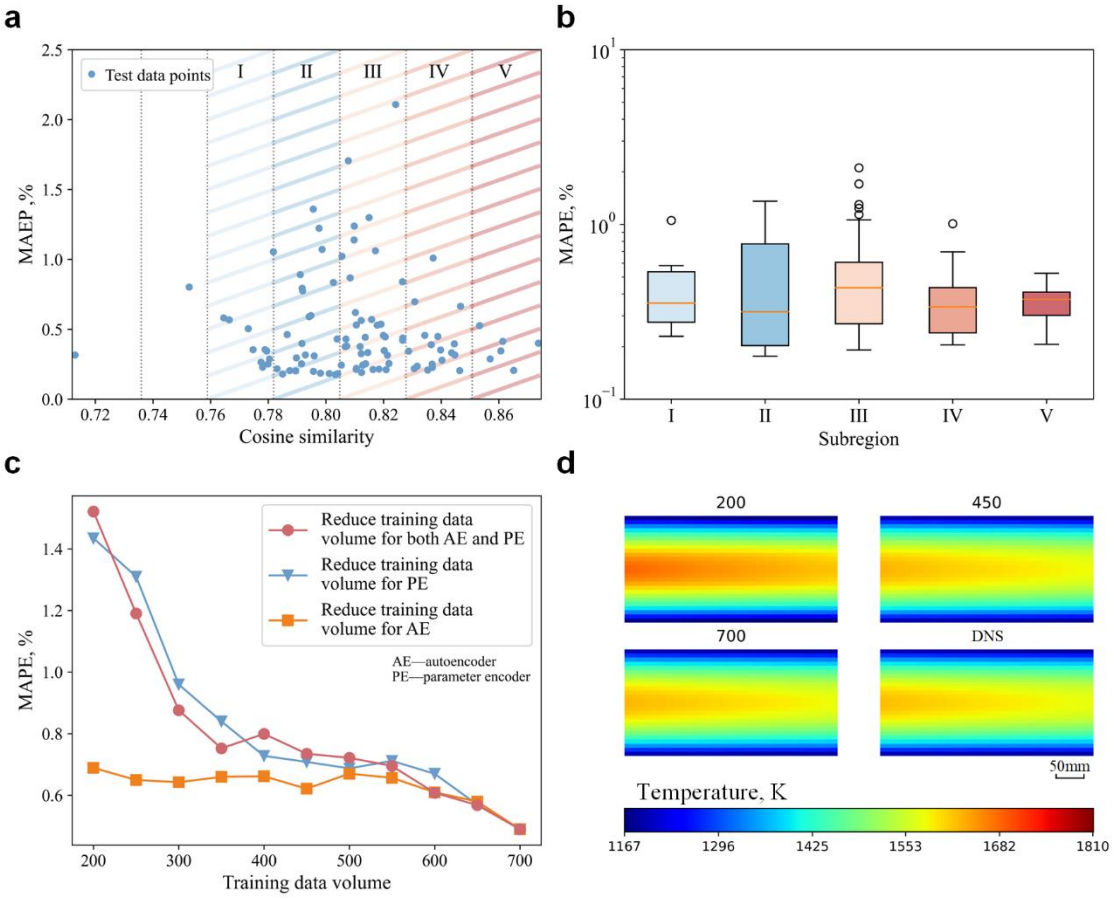

Fig. 6 Experiment results: the effect of training data volume on the prediction results. **a** Mean absolute percent error (MAPE) distribution of 100 test data. The cosine similarity between the test technological parameter set and the whole training technological parameter sets is set as the x-axis. Shadings indicate different sub-regions. **b** Box-whisker plot in five sub-regions in Fig. 6(a), n = 11, 25, 38, 18, 6 independent samples for box 1-5. **c** MAPE results when reducing the training data volume from 700 to 200. Red circle curve: reduce the training data volume of both the autoencoder and parameter encoder; blue triangle curve: reduce the training data volume of the parameter encoder and keep 700 training data of the autoencoder; yellow square curve: reduce the training data volume of the autoencoder and keep 700 training data of the parameter encoder. **d** Comparison of prediction results for different training data volume and DNS result.

Tables

Table 1 The original, mutation and new parameter setting during the optimization.

		Original	Mutation	New
Casting speed(m/min)		1.3	**1.37**	**1.37**
Cooling water flow rate in each cooling zone (L/min)	Zone 1	102	102	**127**
	Zone 2	87	87	**106**
	Zone 3	89	89	89
	Zone 4	85	85	85
	Zone 5	70	70	70
	Zone 6	30	30	30
	Zone 7	77	77	77
	Zone 8	49	49	49
Bold font indicates the parameter that has changed.				