communications biology

Article

https://doi.org/10.1038/s42003-024-07031-6

GoToCloud optimization of cloud computing environment for accelerating cryo-EM structure-based drug design

Toshio Moriya [1,3,4] ✉, Yusuke Yamada[1,2,3], Misato Yamamoto[1] & Toshiya Senda [1,2,4] ✉

Cryogenic electron microscopy (Cryo-EM) is a widely used technique for visualizing the 3D structures of many drug design targets, including membrane proteins, at atomic resolution. However, the necessary throughput for structure-based drug design (SBDD) is not yet achieved. Currently, data analysis is a major bottleneck due to the rapid advancements in detector technology and image acquisition methods. Here we show "GoToCloud", a cloud-computing-based platform for advanced data analysis and data management in Cryo-EM. With GoToCloud, it is possible to optimize computing resources and reduce costs by selecting the most appropriate parallel processing settings for each processing step. Our benchmark tests on GoToCloud demonstrate that parallel computing settings, including the choice of computational hardware, as well as a required target resolution have significant impacts on the processing time and cost performance. Through this optimization of a cloud computing environment, GoToCloud emerges as a promising platform for the acceleration of Cryo-EM SBDD.

Cryogenic electron microscopy (Cryo-EM) single particle analysis (SPA) is the most widely used method for the 3D structural visualization of membrane proteins and large macromolecular assemblies at a resolution where atomic modelling is possible[1]. Although such large molecules are major targets for drug discovery, most of them are difficult to crystallize and therefore not applicable for X-ray crystallography[2–5]. As an alternative for compound screening in structure-based drug design (SBDD), Cryo-EM SPA would be the logical choice, but the throughput required for this purpose has not been achieved. Additionally, in the last few years, it has become possible to acquire a large number of micrograph movies in a short period of time (over 10,000 movies per day) owing to advances in detector technology and image acquisition methods[6–8]. Consequently, data analysis has become a major bottleneck in Cryo-EM SPA, and huge amounts of computational resources are now needed to analyse the massive amounts of data. Therefore, it is critical to establish a computational platform and an efficient Cryo-EM SPA workflow specifically designed for rapid determination of compound-binding protein structures to realize the practical application of Cryo-EM to SBDD.

Researchers in fields that require massive amounts of calculations based on a huge dataset, such as Cryo-EM SPA, are turning their attention to high-performance computing (HPC), especially cloud computing (Cloud)[9–15]. By utilizing Cloud services such as Amazon Web Services (AWS), Google Cloud Platform, and Microsoft Azure, users can perform calculations using virtually limitless computational resources, including storage. These services operate on an on-demand basis, allowing users to quickly and inexpensively procure the required amount of computational resources as needed, in contrast to computational resources that must be acquired and maintained on-premise. With Cloud services, it is relatively easy to improve cost performance without specialized knowledge of HPC or experience in operating computing facilities over a long period of time.

Particularly, AWS offers a service called AWS ParallelCluster (pcluster), which allows users to build, configure, and manage HPC clusters on an on-demand basis with a relatively simpler procedure than designing and building a cluster from scratch by themselves. The service automatically and securely distributes the necessary computational resources for the applications based on user-specified configurations. AWS supports over 200 types of virtual machines, called "Amazon Elastic Compute Cloud (EC2) instances". Each of the EC2 instance types has a different configuration of the central processing unit (CPU), memory, graphics processing unit (GPU), storage, and networking resources. Notably, different EC2 instance types can coexist in the compute nodes (or simply "nodes") of a single pcluster instance by defining multiple queues of the job scheduler

[1]Structural Biology Research Center, Photon Factory, Institute of Materials Structure Science, High Energy Accelerator Research Organization (KEK), Tsukuba, Japan. [2]Department of Materials Structure Science, School of High Energy Accelerator Science, The Graduate University of Advanced Studies (Sokendai), Tsukuba, Japan. [3]These authors contributed equally: Toshio Moriya, Yusuke Yamada.[4]These authors jointly supervised this work: Toshio Moriya, Toshiya Senda. ✉e-mail: toshio.moriya@kek.jp; toshiya.senda@kek.jp

(e.g. Slurm) in the associated configuration file. Therefore, it is possible for users to build customized HPC clusters having a variety of hardware configurations. Since AWS continuously introduces the latest hardware in a timely manner, by utilizing the pcluster mechanism effectively, one can try out the latest computing hardware promptly. In this way, scientists can design and build any number of HPC clusters with various hardware configurations that meet their current needs and priorities and then immediately utilize them. However, this task is still difficult for the majority of structural biologists.

Cloud applications for Cryo-EM SPA have already been attempted. Cianfrocco et al. developed "cryoem-cloud-tools"[16,17], an analysis environment for running SPA jobs on AWS Cloud, which supports multi-node processing for each SPA job. As a similar system, "ScipionCloud"[18], which utilizes a single GPU-enabled computer for each SPA job, has also been developed. The "Stion" system (https://aws.amazon.com/jp/blogs/hpc/stion-a-saas-for-cryo-em-data-processing-on-aws/)[19] also assumes the use of a single GPU-enabled computer for each SPA job, and was designed as a training environment for SPA beginners rather than a resource for practical data analysis. Recently, a practical analysis environment for "CryoSPARC"[20] using pcluster has also been announced (https://assets.thermofisher.com/TFS-Assets/MSD/Reference-Materials/pharma-cryosparc-wp0028.pdf). However, the implementation of CryoSPARC currently supports only multi-GPU processing and multi-threading on a single node and does not support multi-node processing. This results in a significant constraint—namely, the scalability is limited by the available machine types (e.g. the number of GPU cards on a single EC2 instance type is 1 GPU, 4 GPU, 8 GPU, or 16 GPU cards on AWS). Thus, the strengths of pcluster are not fully exploited.

To address the issues of the previously developed AWS cloud-based Cryo-EM SPA environments, we developed "GoToCloud", a practical platform that achieves a performance level suitable for industrial applications using the multi-node processing of pcluster (Fig. 1a). The aim is to establish a practical computational infrastructure for the cryo-EM field using the latest cloud computing technologies through a publicly available service for both academia and industry. Our approach can centralize the technical aspects of maintaining and optimizing the data processing software, and optimizing the use of AWS resources, letting the users focus on processing their data. We conducted benchmark tests using practical datasets to verify the scalability and cost-effectiveness of the GoToCloud platform. Based on the benchmark results, we determined an optimal balance between processing speed and cost. In addition, by performing structure determination at a resolution higher than 2 Å, we evaluated the relationship between resolution improvement and cost.

Results

Design philosophy of the GoToCloud platform

A major challenge of using AWS is that expertise in various AWS-specific services is required to ensure optimal system configuration and security. AWS has a large set of managed services that allow many advanced features to be used easily, providing flexibility and scalability to accommodate various purposes, and the number of such services is still increasing rapidly. Ironically, this strength also requires users to be familiar with the AWS system to design and build a platform that meets their own needs. To achieve specialized goals, a system must be built with the optimal combination of multiple managed services. In addition, in fields such as Cryo-EM structural biology, where dataset size and analysis computation are enormous, it is necessary to guarantee minimum storage size, storage access speed, memory capacity, and processing speed that meet practical requirements. Furthermore, the selection of a minimum set of EC2 instance types from over 200 types while allowing sufficient flexibility for a given purpose is indispensable, making the barrier even higher. Because of these difficulties, it is evident that building a domain-specific platform is an extremely heavy burden for most scientists.

To address these issues, we designed and developed the "GoToCloud" platform specialized for data analysis in Cryo-EM SPA (see also 'sections 1.1–1.4' in the Supplementary Information). The design mainly targeted those researchers in the field of structural biology, most of whom are not familiar with the AWS services and implementation details of parallel computing supported by required analysis software packages. Accordingly, we aimed to automate the design and construction of pcluster instances as a ready-to-use Cryo-EM SPA computational platform by (1) pre-selecting machine specifications (EC2 instance types), (2) pre-installing dependent components (e.g. pcluster library, JSON processor (jq), and JavaScript runtime environment (Node.j)), and (3) pre-selecting and pre-installing analysis software with optimal compilers with an optimal set of the compile options (Fig. 1a). An additional reason for pre-installing the analysis software was to provide a set of the latest software packages within a secure environment in a timely manner, so that users would not be required to update any of these packages by themselves. We adopted RELION[21,22] as the main SPA software because of its multi-node computing support with a message-passing interface (MPI). Multiple RELION executables were built with various compilers with various sets of compiler flags and pre-installed on the platform, to provide an optimal executable for each RELION job type.

For GoToCloud, we adopted a system architecture that maps the relevant real-world objects to AWS-managed services by considering a common use case in Cryo-EM SPA (Fig. 1b). We assume that there are multiple research groups, and that the analysis data is held confidentially within each research group. Therefore, we define that a security unit corresponds to each research group and assume that each of the groups has its own AWS account (AWS account (User)). This way, we can utilize the robust security services already provided on an AWS per-research-group basis. Another crucial component in this architecture is an AWS account maintained by the GoToCloud management group (AWS account (GoToCloud)). The management account has an Amazon Elastic File System (EFS) as storage to be shared by all users. The related software packages are pre-installed in this shared EFS. With this architecture, GoToCloud can allow all users to instantly use the latest analysis software environment maintained by the Cryo-EM SPA experts.

To build the GoToCloud platform, various AWS-managed services must be built and set up (Fig. 1b), which can be a cumbersome task to perform manually. Therefore, to allow users to perform these preparations with as few steps as possible, we have developed a set of scripts called GoToCloud scripts (GTC scripts)[23], which enables the construction of a ready-to-use Cryo-EM SPA computing platform in just three steps (Supplementary Fig. S1). In the third step, users can remotely access the NICE-DCV remote desktop environment on the head node of the constructed pcluster instance through a WEB browser, and start analysis immediately using the graphic user interface (GUI) of the analysis software (e.g. RELION4.0[22] and UCSF Chimera[24]) as before (Supplementary Fig. S2). These scripts are also stored in the shared EFS so they can be accessed by the users of all AWS accounts where the GoToCloud platform is set up. A manual procedure for building a GoToCloud platform instance typically takes approximately half a day to a full day even for the developers who are well familiar with the steps. With the GTC scripts and online documentation of step-by-step instructions (see the 'Code availability section'), this task can be completed in 30–60 min for users at any level of familiarity with the AWS system, making the GTC scripts highly valuable. Moreover, since most of the time spent is just waiting for the command executions, the actual time spent by the user is even shorter. In addition, by carefully designing the specifications of the GTC scripts, the possibility of users making mistakes has been minimized. The detailed descriptions of the GTC scripts are given in the 'Methods' section (see also 'section 1.4' of the Supplementary Information).

Benchmark tests with a realistic dataset

To verify the effectiveness of the GoToCloud platform, we conducted benchmark tests using a realistic dataset (EMPIAR-10581[25] in Supplementary Table S1) to validate scalability and cost-performance because it is essential to achieve high practicality by selecting the optimal machines and parallel computing parameter settings for each processing job. In this

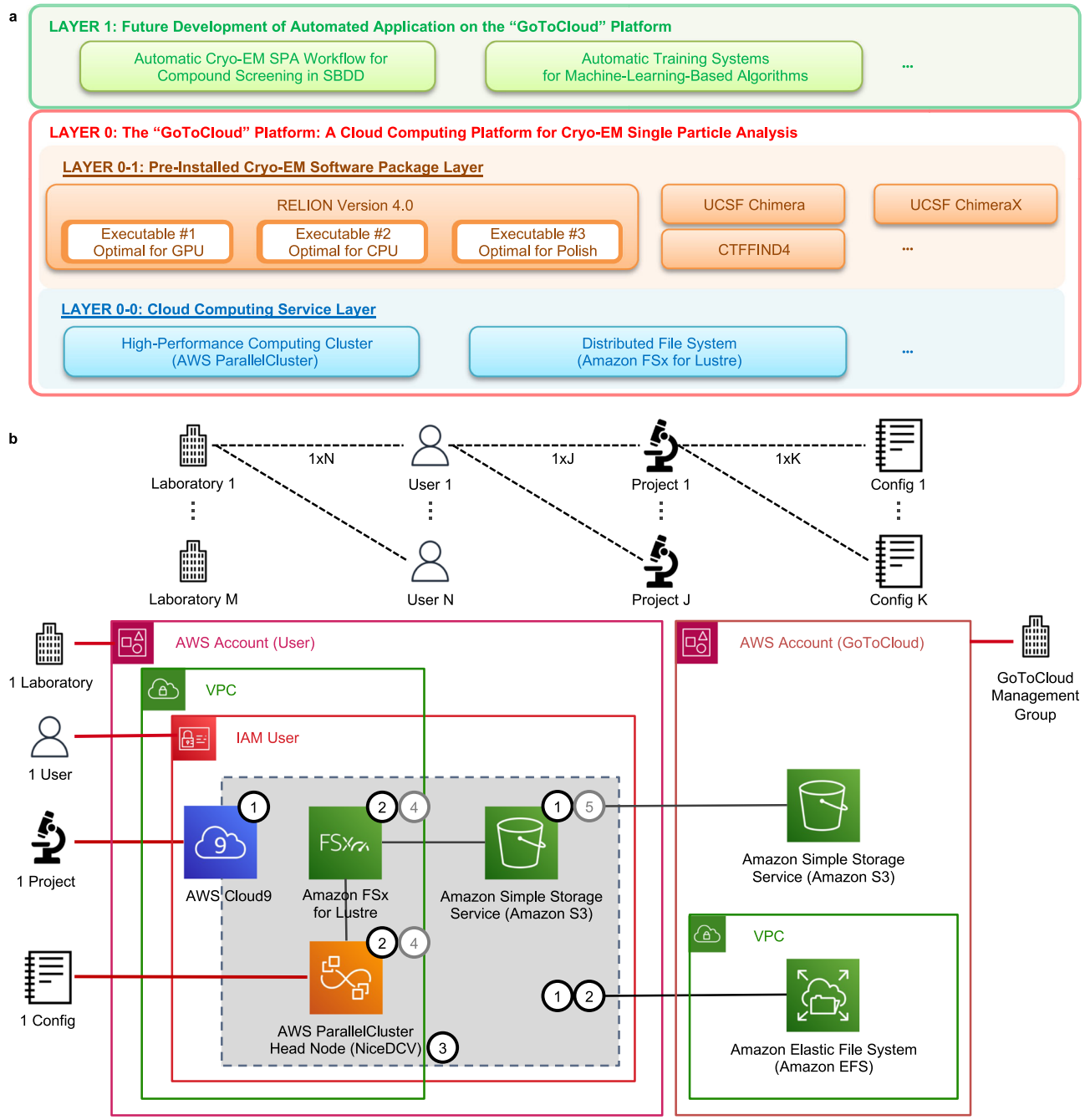

Fig. 1 | The GoToCloud platform, system mapping and architecture. A diagram of the "GoToCloud" platform and system mapping between the relevant real-world objects appears in a common practice of Cryo-EM SPA and AWS-managed services. **a** Schematic diagram depicting the conceptual layers and examples of their components in the GoToCloud platform. **b** Schematic diagram of the system mapping used to design the GoToCloud architecture. A security unit corresponds to each research group having its own "AWS account (User)". The "AWS account (GoToCloud)" is maintained by the GoToCloud management group. The management account has shared "Amazon EFS" storage where the GoToCloud scripts for building and setting up the computational platform instance are stored and where related software packages are pre-installed. By sharing this storage, the GoToCloud can allow all users to instantly use the latest analysis software environment maintained by the Cryo-EM SPA experts in the GoToCloud management group. The circled numbers indicate the step numbers of the GoToCloud scripts which build, delete, or set up each associated AWS managed service(s) (Supplementary Fig. S1).

benchmark, we used Nitrite reductase (NiR), which is a small protein with C3 symmetry and a total molecular weight of 110 kDa. The original work achieved a 2.85 Å resolution[26]. Here, we targeted the 3D Refinement (Refine3D), 3D Classification (Class3D), 2D Classification (Class2D), and Bayesian Polishing (Polish) jobs of RELION4.0[27]. These processes are the most important but time-consuming processes in the Cryo-EM SPA workflow. The benchmark tests were conducted using on-demand and spot instances with AWS ParallelCluster v3.0.3 in the US East (Northern Virginia) AWS Region. Two EC2 instance-type groups, G5 and G4dn, were used for the processes that support GPU and C6i was used for the other processes. Each of the G5 and G4dn instance type groups has multiple instance sizes equipped with the same number of GPUs (i.e. 1, 4, and 8) but a different number of virtual CPUs (vCPUs). Therefore, a single EC2 instance type among those with a particular number of GPUs that achieved the best performance, meaning the best balance between the numbers of GPUs and vCPUs, in preliminary trial runs was selected for each RELION job type.

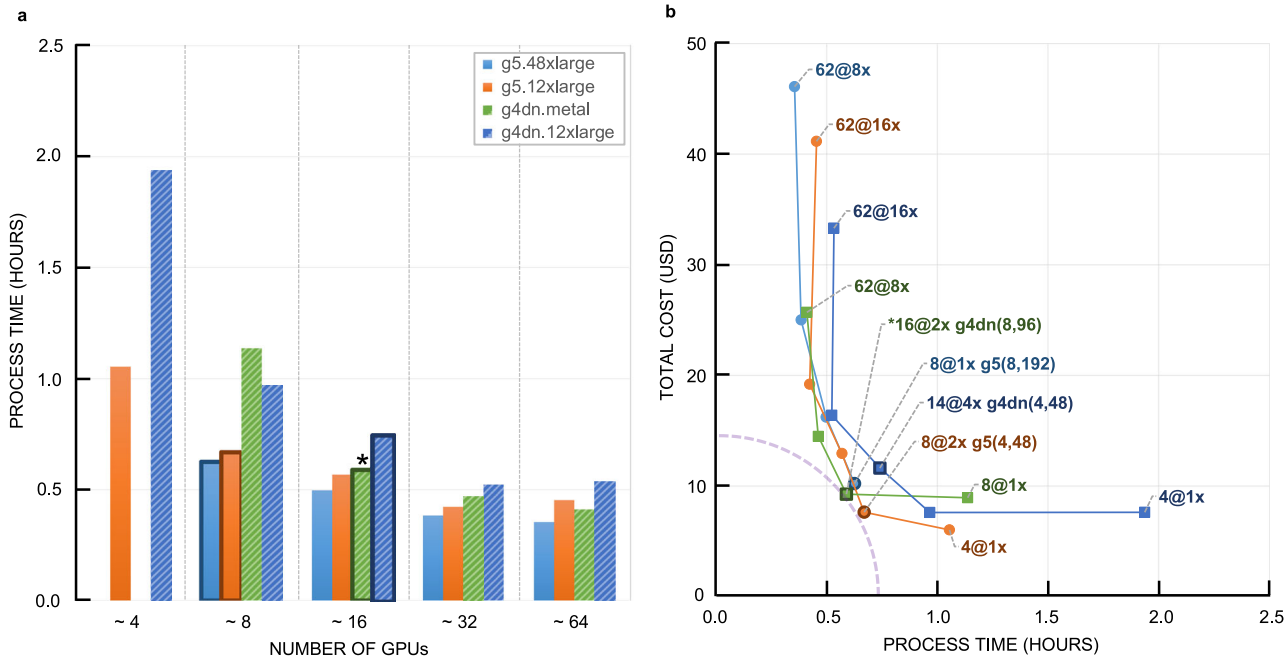

Fig. 2 | NiR dataset benchmark result of 3D refinement (Refine3D). The bar graph and scatter plot of the NiR dataset benchmark result of Refine3D using the RELION executable EXE01_GPU in Supplementary Data 1. The data of AWS EC2 instance types of g5.48xlarge (light blue) and g5.12xlarge (orange) from the G5 instance type group (solid bars or circles), and g4dn.metal (green) and g4dn.12xlarge (blue) from G4dn (striped bars or rectangles) are shown. **a** Bar graph for the scalability assessment showing the processing time in hours relative to the total number of GPU cards used for a single Refine3D job. **b** Scatter plot for investigation of the virtual cluster configuration achieving the optimal cost performance. The horizontal axis is the processing time in hours and the vertical axis is the total cost of processing in US dollars, where the aspect ratio is 10 US dollars per 0.5 h. The dashed arc indicates the distance of this globally optimal configuration from the origin. For the optimal configuration in each EC2 instance type, the format of the marker label is "[the total number of GPUs used]@[the number of nodes]x [EC2 instance type group]([the number of GPUs per node],[the number of vCPUs per node])". For other non-optimal configurations, the format is "[the total number of GPUs used]@[the number of nodes]x". To clarify the associations, the configurations using the same EC2 instance type are connected by a line. In both panels, the optimal configuration in each EC2 instance type is outlined. The asterisk (*) indicates the globally optimal configuration. The raw data of all AWS EC2 instance types used in this benchmark, including g5.4xlarge and g4dn.4xlarge, are provided in the Refine3D sheet of Supplementary Data 2.

The P4 (p4d.24xlarge) and P5 (p5.48xlarge) instances were also considered, but we chose the G4dn and G5 instance types over P4 and P5 primarily because of their flexibilities in GPU card numbers and hourly rates that allow us to optimize the number of GPUs (i.e. less than eight GPU cards) and so reduce the cost for each RELION job type. While P4 and P5 instance types support only the eight GPU configurations, G4dn and G5 instance types offer configurations with 1 GPU, 4 GPU, and 8 GPU cards. Additionally, the hourly rates for P4 (32.7726 USD) and P5 (98.32 USD) instances are significantly higher than those for G4dn (USD 7.824 for g4dn.metal) and G5 (USD 16.288 for g5.48xlarge) instances with eight GPU cards. Therefore, finer adjustments for cost performance are achievable with G4dn and G5.

All the EC2 instance types used in the benchmark are selectable within a pcluster instance built by the GTC scripts and their specifications are summarized in Supplementary Table S2. Three types of pre-installed RELION4.0 executables were prepared by compiling the source code using the GCC 9.3.0 and 9.4.0 compiler and the classic compiler in the Intel® oneAPI Base Toolkit 2022.1.2.146 and Intel® oneAPI HPC Toolkit 2022.1.2.117 (Intel Inc.) with different compilation settings. The details of the executables their compilation settings, and their applied RELION job types are summarized in Supplementary Data 1. The input parameters for each processing job that strongly affect processing speed are summarized in Supplementary Table S3. The results of all benchmark tests are provided in Supplementary Data 2. As a reference for readers, Supplementary Data 2 also contains a breakdown of all the associated costs, as well as a "grand total cost", including the storage costs. Since the pricing of the S3 and EFS provided by AWS is listed as a monthly rate per GB (USD/GB/month), the costs associated with the S3 and EFS were extracted from the billing statement of our AWS account for the month (February 2023) when the benchmark tests were conducted (provide in the "Pricing" sheet). For the calculation of the grand total cost for each RELION job execution, these storage costs in the month are further multiplied by the ratio of the processing time (in hours) relative to a month.

Results of the 3D refinement (Refine3D) benchmark test

The results of the Refine3D benchmark test are summarized in Fig. 2. The key goals of GoToCloud are to improve the processing speed and optimize cost-effectiveness. For this, along with a bar graph (Fig. 2a) of the type frequently used for benchmark tests, we devised a scatter plot with total processing time on the *x*-axis and total processing cost on the *y*-axis (Fig. 2b). A near-ideal scalability, where increasing the number of computer resources improves the processing speed without any significant addition of costs, is represented by a horizontal line in this plot. The worst scalability, where increasing the number of computer resources does not enhance processing speed at all but only the cost increases, is represented by a vertical line. The ideal cost performance should be no cost and no processing time, which corresponds to the origin of the plot. Therefore, we define that the parallel computing settings associated with the closest point to the origin achieve the optimal cost performance where the balance between processing speed and cost is the best.

The scalability of Refine3D was saturated with relatively few GPUs: 8 GPU and 16 GPU configurations of the G4dn and G5 instance types, respectively (Fig. 2a). Sixteen GPUs with two nodes of g4dn.metal (eight GPUs) showed the optimal cost performance (closest to the origin) (Fig. 2b). In addition, eight GPUs with two nodes of g5.12xlarge (four GPUs) showed a similar cost performance to the optimal (second closest to the origin). Notably, up to the saturation point of the scalability, configurations with G5 instance types achieved nearly the same processing speed and total cost using half the number of GPUs with the G4dn instance types. This means that the single G5 GPU showed twice the performance in terms of

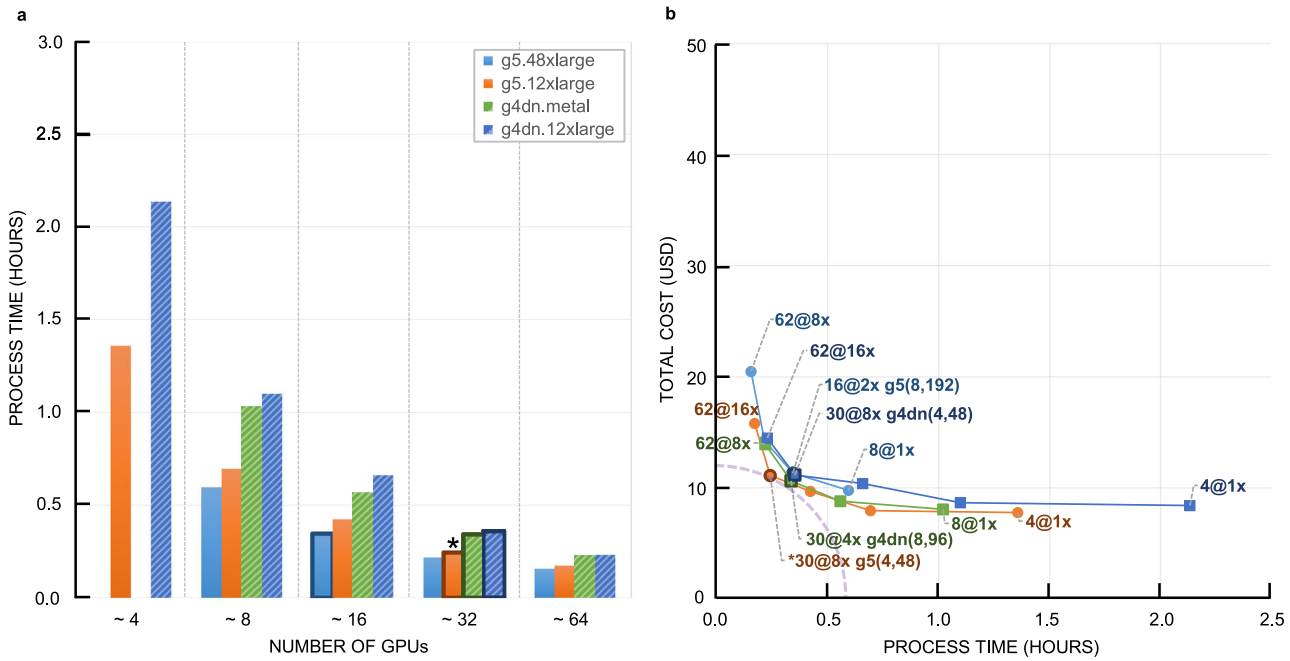

Fig. 3 | NiR dataset benchmark result of 3D classification (Class3D). The bar graph and scatter plot of the NiR dataset benchmark result of Class3D. The data obtained with the same RELION executable and AWS EC2 instance types as in Fig. 2 are shown. **a** Bar graph for assessment of the scalability. **b** Scatter plot for investigation of the virtual cluster configuration achieving the optimal cost performance. The axes, colours, bars, maker symbols, and marker labels are the same as in Fig. 2. The raw data of all AWS EC2 instance types used in this benchmark, including g5.4xlarge and g4dn.2xlarge, are provided in the Class3D sheet of Supplementary Data 2.

processing speed compared to the single G4dn GPU, but the price per hour with G5 was approximately twice that with G4dn (e.g. 16.288 USD for g5.48xlarge and 7.824 USD for g4dn.metal; both with eight GPUs), resulting in nearly the same total cost. Comparing CPU-optimized executions to GPU-optimized ones demonstrated that both computation time and cost were greatly increased (Supplementary Fig. S3a).

Results of the 3D classification (Class3D) benchmark test

The scalability of Class3D showed a similar trend to that of Refine3D, but the Class3D scalability was saturated with a larger number of GPU cards: 16 and 32 GPU configurations of the G4dn and G5 instance types, respectively (Fig. 3a). Again, the G5 GPUs achieved nearly the same processing speed and total cost with half the number of G4dn GPUs up to the saturation point of the scalability. 30 GPUs with eight nodes of g5.12xlarge (four GPUs) showed the optimal cost performance (Fig. 3b). The comparison of CPU-optimized executions relative to GPU-optimized ones showed again that both computation time and cost were greatly increased (Supplementary Fig. S3b).

Results of the 2D classification (Class2D) benchmark test

The Class2D with the expectation maximization (EM) algorithm showed a near-ideal scalability proportional to the number of GPUs up to ~32 (Fig. 4a). Thirty GPUs with eight nodes of g4dn.12xlarge (four GPUs) instance type resulted in the optimal cost performance (Fig. 4b). Interestingly, unlike Refine3D and Class3D, G4dn showed significantly better cost performance than G5 with Class2D. The comparison of CPU-optimized executions relative to GPU-optimized ones showed that computation time decreased, and while cost still increased, the amount of increase was much smaller than that in the cases of Refine3D and Class3D (Supplementary Fig. S3c).

An additional benchmark test of the 2D classification job was conducted using the variable-metric gradient descent with the adaptive moments algorithm option (Class2D-VDAM)[22]. The original report by the developer has shown that the Class2D-VDAM is faster than Class2D with the EM algorithm[28,29], especially for large datasets. Since Class2D-VDAM supports multiple GPU cards but not MPI, the effect of the number of GPU cards used within a single EC2 instance type was evaluated. The preliminary trial runs showed that the number of threads should be the same as the number of GPU cards used for the processing job to achieve the best performance. The results demonstrated that more GPU cards achieved shorter processing time and cheaper cost (Supplementary Fig. S4). The scalability was saturated at four GPUs (Supplementary Fig. S4a), and the configuration of four GPUs with one node of g5.12xlarge (four GPUs) achieved the best cost performance (Supplementary Fig. S4b). The best cost performance of Class2D-VDAM was not significantly different from that of Class2D with the EM algorithm because the distances of the associated optimal setting points from the origin were close to each other.

Results of the Bayesian polishing (Polish) benchmark test

With polish, only C6i was used since this job type does not support GPU. Polish exhibited almost perfect scalability proportional to the number of nodes, up to 4 (Fig. 5). Beyond this, although the total processing time decreased, the total cost substantially increased. The four nodes of c6i.32xlarge (256 vCPU) with four MPIs per node gave the optimal cost performance. Interestingly, in settings with four or fewer nodes, processing time decreased as the number of MPIs per node increased, but as the number of nodes increased, the processing time difference depending on the number of MPIs per node disappeared.

Results of the benchmark tests with spot instances

On AWS, the user can request an unused EC2 instance as a spot instance with a significantly reduced price. Comparisons of spot instance executions relative to on-demand executions in NiR dataset benchmarks showed that spot instances had superior cost performance without any influence on the execution time (Supplementary Fig. S5). During the benchmark executions, the weekly average of the discount rates of spot instances was 68% for g4dn.metal (Refine3D), 70% for g5.12xlarge (Class3D), 70% for g4dn.12xlarge (Class2D), and 60% for c6i.32xlarge (Polish). The average values were obtained from the "Spot Instance pricing history" page in the AWS management console. The mean and standard deviation of cost reduction percentage for Refine3D, Class3D, Class2D, and Polish were 66 ± 4.1%, 70 ± 0.3%, 70 ± 0.3%, and 56 ± 7.1%, respectively, where each reduction percentage is calculated for a pair of on-demand and spot instance with the same parallel computing settings using the discount rates above.

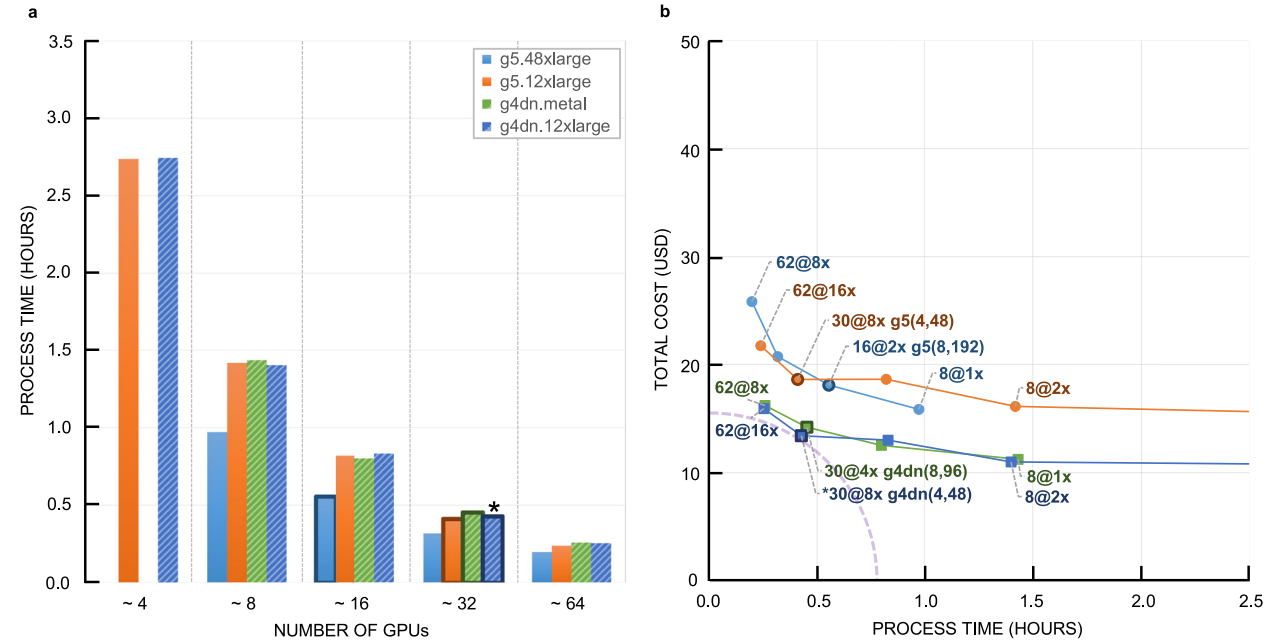

Fig. 4 | NiR dataset benchmark result of 2D classification (Class2D). The bar graph and scatter plot of the NiR dataset benchmark result of Class2D. The data obtained with the same RELION executable and AWS EC2 instance types as in Fig. 2 are shown. **a** Bar graph for assessment of the scalability. **b** Scatter plot for investigation of the virtual cluster configuration achieving the optimal cost performance. The axes, colours, bars, maker symbols, and marker labels are the same as in Fig. 2. The raw data of all AWS EC2 instance types used in this benchmark, including g5.xlarge and g4dn.2xlarge, are provided in the Class2D sheet of Supplementary Data 2.

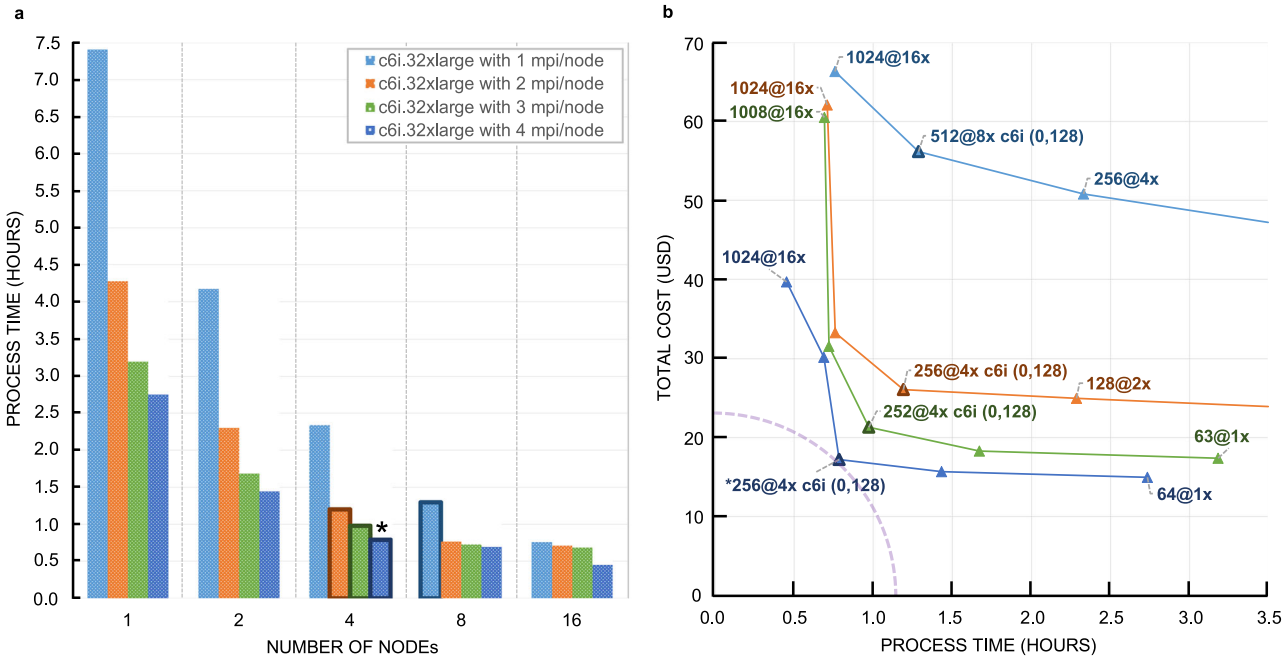

Fig. 5 | NiR dataset benchmark result of Bayesian polishing (Polish). The bar graph and scatter plot of the NiR dataset benchmark result of Polish using the RELION executable EXE03_CPU in Supplementary Data 1. With the AWS EC2 instance type of c6i.32xlarge from the C6i instance type group (dotted bars or triangles), the MPI processes per node (MPIs/node) were varied: 1 (light blue), 2 (orange), 3 (green), and 4 (blue). Settings of more than four MPIs/node resulted in memory capacity issues. Only physical CPUs (i.e. half of vCPUs) were used in the Polish jobs. **a** Bar graph for the scalability assessment showing the processing time in hours relative to the number of nodes used for a single Polish job. **b** Scatter plot for investigation of the virtual cluster configuration achieving the optimal cost performance. For the optimal configuration in each MPI/node-set, the format of the marker label is "[the total number of vCPUs used]@[the number of nodes]x [EC2 instance type group]([the number of GPUs per node],[the number of vCPUs per node])". For other non-optimal configurations, the format is "[the total number of vCPUs used]@[the number of nodes]x". The other attributes of the panels are the same as in Fig. 2. The raw data are provided in the Polish sheet of Supplementary Data 2.

Results of a further optimization for the G5 instance types

A further optimization for the G5 instance types equipped with NVIDIA A10G Tensor Core GPUs was attempted. We built another RELION executable using the cmake configuration optimal for the compute capability of the CUDA architecture of A10G (i.e. "-DCUDA_ARCH = 86"), in addition to the one which is optimal for the G4dn with NVIDIA T4 Tensor Core GPU (i.e. "-DCUDA_ARCH = 75"), and conducted benchmarks for Refine3D, Class3D, and Class2D using the G5 instance types. The

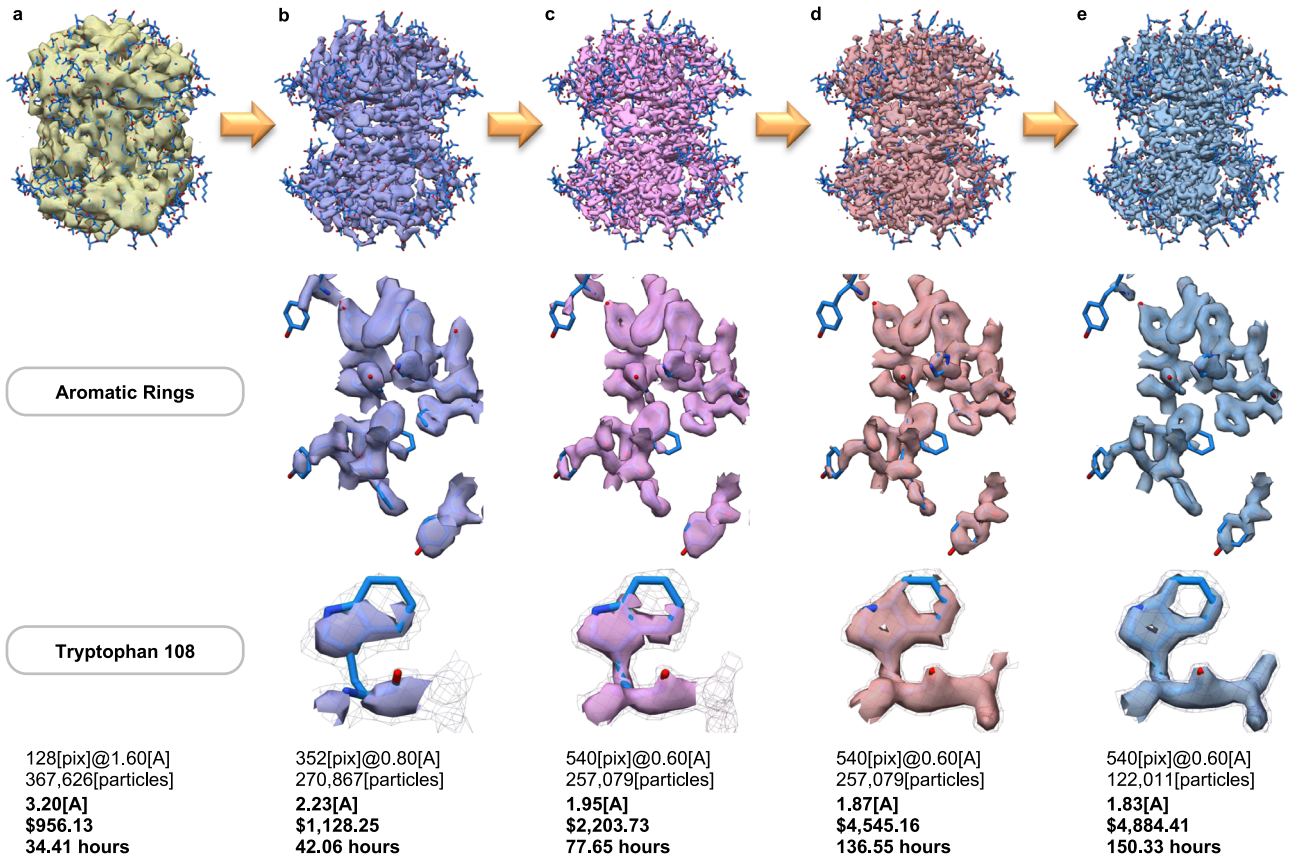

Fig. 6 | The relationship between resolution improvement and cost. The Cryo-EM SPA image processing results of the streptavidin dataset (EMPIAR-10641) for assessment of the relationship between resolution improvement and cost. **a** Results of the stack cleaning process using Class2D and Class3D with a binning factor of 4. **b** The results after reducing the binning factor to 2 and selections by Class3D. **c** The point where the resolution reached higher than 2.0 Å with the 2nd CTF refinement during the repeated cycles of CTF refinement and Polish. **d** The result at the end of the 5th cycle of CTF refinement and Polish. **e** The final map was obtained by refining the homogeneity of the particle stack using no-alignment Class3D.

results showed clear improvements in execution times and costs for Refine3D, a slight improvement for Class3D, and no improvement for Class2D (Supplementary Fig. S6). The improvements become more pronounced as the number of nodes increases.

Relationship between cost and resolution improvement in the Cryo-EM SPA workflow

We also investigated the relationship between cost and resolution improvement by determining an atomic resolution structure. The motivation of this case study was to find how the processing time and cost of a practical usage increases when one tries to improve resolution by doing additional processes on the GoToCloud platform and if the relationships between the processing time and cost relative to the resolution improvement are more like linear or exponential using a typical SPA workflow. For this verification, RELION3.1.2[21] was used. The analysis was performed using on-demand usage in the Asia Pacific (Tokyo) Region. We chose a streptavidin dataset (EMPIAR-10641[30] in Supplementary Table S1), since the associated EMDB map has a resolution of 1.93 Å (EMD-30913[31]), which is higher than the 2.0 Å preferred for drug discovery, making it a practical dataset for the SBDD application. Streptavidin, a tetrameric protein with the point group symmetry of D2, is a challenging target because of its small molecular weight (~53 kDa). With GoToCloud, a map with a resolution of 1.83 Å was obtained. We extracted jobs directly related to the final map and provided the parallel computing settings, input parameters that strongly influence processing speed, and important output values in Supplementary Data 3.

The resolution along with the accumulated total processing time and total costs at the key steps of Cryo-EM SPA are summarized in Fig. 6. The resolutions improved from 3.20 Å to 2.23 Å, 1.95 Å, 1.87 Å, and finally 1.83 Å. The accumulated total processing times were 34.41 h, 42.06 h, 77.65 h, 136.55 h, and 150.33 h, while the associated total costs increased from 957 USD to 1129 USD, 2204 USD, 4546 USD, and 4885 USD. The total processing time and cost were approximately doubled from the second step to the third to obtain a resolution improvement of 0.28 Å, and from the third step to the fourth for a resolution improvement of 0.08 Å due to repeated use of the computationally intensive Polish. As evident from the comparison of the appearance of the holes in the aromatic ring side chains, the improvements of 0.28 Å, 0.08 Å, and 0.04 Å in the resolution range of around 2.0 Å greatly improved the appearance of the side chains.

Discussion

In the current study, we demonstrated the validity of GoToCloud in Cryo-EM SPA by performing several benchmark tests. The results demonstrated that our optimization of parallel computing settings had a significant impact on both processing time and cost performance, so it is important to use the optimal settings for each combination of computer hardware and image processing job. Particularly, conducting a long-term operation with an automated workflow using suboptimal parallel computing settings can result in significant losses of time and money. From the processing of the streptavidin dataset, it is evident that the total processing time and cost increase exponentially as the resolution becomes higher. This result indicates that one should consider omitting jobs in the later stage of the SPA processing if the resolution sufficient for research purposes is achievable without them. Specifically, employing a smaller number of Polish cycles is effective in reducing the processing time and cost. Therefore, in the SBDD application, it is suggested that

setting a minimum necessary resolution while considering the corresponding appearances of the side chains is crucial to achieve a shorter processing time and cost reduction.

We reported here the closest point to the origin as the optimal setting where the processing time and cost are most well balanced. However, the meaning of "optimal" can be changed for each Cryo-EM SPA project. As expected, a clear common trend of cost performance relative to the increase of computer resources can be seen from the graphs of all benchmark tests. For a given combination of sample nature, imaging quality, dataset size, RELION job type, and parallel computing settings including specific computer hardware (i.e. for a given EC2 instance type), there are apparently ultimate limitations for the attainable shortest processing time and lowest cost. Starting from a small number of computer resources (right side of each curve), the curve remains roughly horizontal, and its *y*-axis value defines the lowest attainable processing cost of a particular combination. Then, the curve starts to go up from right to left and finally asymptotes a vertical line and its *x*-axis value defines the shortest attainable processing time. Therefore, for a project where cost reduction is more important than minimizing the processing time, one should choose the point nearest the leftmost position on this horizontal part of a curve. Conversely, when a decrease in processing time must be prioritized over a decrease in cost, one may want to choose the point nearest the bottom of the vertical part of a curve. A general guideline is to select a point on a sloping portion of a curve based on the required weight ratio between processing time and cost for a given project. To this end, the results of this study can be useful in optimizing the parallel computing settings for a fully automated Cryo-EM SPA workflow aiming at the practical industrial application of SBDD.

An important consideration is the general applicability of the proposed optimal parallel computing settings for each RELION job type and/or other datasets. Our benchmarking results, based on a single system and dataset, serve as an illustrative example specific to the configuration used in this study. While these results offer valuable insights into optimizing parallel computing settings on the GoToCloud platform, they are primarily intended to help users customize these settings according to their research priorities, as discussed above. Unlike on-premise computing resources, which are often not publicly accessible and likely offer limited availability to external users, the AWS cloud computing service is publicly available. This accessibility makes it easier for researchers in both academia and industry to optimize parallel computing settings on the same platform according to their needs by referring to our benchmark results. Nevertheless, our benchmarking results should be viewed as an example rather than a comprehensive validation. Therefore, outside the GoToCloud platform, we encourage readers to conduct their own benchmarks on their specific systems, to achieve optimal performance. Additionally, the transferability of these results to other datasets is another important consideration, and this should be explored in future work to fully automate the cryo-EM SPA workflow.

By leveraging the benefits of the Cloud service and our robust benchmark testing, GoToCloud has provided a solution for the bottleneck in the current data analysis practice of Cryo-EM SPA. Additionally, because of our shared EFS, all users connected to GoToCloud can always use the latest software in a secure analysis environment, even if the analysis software is frequently updated. This is particularly true for fields where the methodologies are still immature but growing fast, as often seen in academia. Therefore, it is expected that similar data analysis platforms will soon be constructed on Cloud services in more fields. At that time, the design philosophy of GoToCloud presented here should provide sound guidance.

We are now ready to develop a fully automated workflow for data processing of the highly demanded Cryo-EM SBDD on top of the GoToCloud platform to accelerate drug design significantly by processing all datasets simultaneously with large-scale parallelization using the optimal settings. As the processing result of the streptavidin dataset indicated, an important issue for automation is to define the stopping condition of the workflow to achieve a high level of practicality which should be addressed in this future study. Furthermore, by utilizing the effectiveness of the Cloud service, GoToCloud can be extended to a more advanced system that supports the Internet of Things (IoT) for this analysis method, which connects multiple Cryo-EM facilities to the Cloud through the Internet, and developmental environments for machine learning (e.g. Amazon Rekognition and Amazon SageMaker) and quantum computing technology (e.g. Amazon Braket)-based algorithms for more advanced automation and acceleration. Thus, GoToCloud can be an essential standard platform for Cryo-EM SPA.

Methods

GoToCloud system mapping

In the GoToCloud architecture, the real-world objects relevant to a common Cryo-EM SPA practice are mapped to AWS-managed services as shown in Fig. 1b. As mentioned in the 'Results' section, multiple research groups are assumed to have their own AWS accounts to utilize the robust security services already provided by AWS on a per-research-group basis. There are multiple users within each research group who perform analysis. In GoToCloud, a user is mapped to an AWS identity and access management (IAM) user. Each user is responsible for multiple projects and a project is associated with a cloud-based integrated development environment (IDE) called AWS Cloud9 (Cloud9). The raw input dataset analysed in each project is expected to be processed using various analysis methods. Therefore, multiple configuration files that define the analysis environment, such as the hardware specifications of the virtual cluster and the required set of software optimized for a given analysis method, can be created for each project. To make it unnecessary for users to maintain these software environments, we developed GTC scripts (see the next section). By placing these scripts in the shared EFS of the GoToCloud management group's account, all the GoToCloud users can instantly use the latest analysis software environment, which is maintained by the Cryo-EM SPA experts.

GTC scripts

By using GTC scripts[23], a user can construct a ready-to-use Cryo-EM SPA computing platform in just three steps without any input parameters (Supplementary Fig. S1). To do so, a user must first create a Cloud9 for the project. Only two input parameters are required: (1) the project name and (2) the network (VPC). Then, the user executes GTC scripts in the Cloud9 terminal. The two conceptual layers of the GoToCloud platform in Fig. 1a are constructed and set up with the Steps 1 and 2 GTC scripts. The Step 1 GTC script prepares for the generation of pcluster instances. This script performs the setup for various AWS services and components altogether (see 'section 1.4' in the Supplementary Information). A project-specific Amazon Simple Storage Service (S3) bucket is generated in this step, and the input dataset is uploaded to the bucket. The Step 2 GTC script creates a new pcluster instance, which consists of a head node, compute nodes, and a high-performance Lustre parallel file system provided by the service called Amazon FSx for Lustre (Lustre). This script automatically mounts the shared EFS on the head node of the constructed pcluster instance, and then sets up the system environment for a selected set of the software packages which have been pre-installed in the shared EFS, such as RELION3.1.2[21], RELION4.0[22], crYOLO[32], CTFFIND4[33], and UCSF Chimera[24]. RELION5.0-beta (an online release announcement in October 2023) has been also pre-installed and can be used already in the GoToCloud platform but this version is not set to default currently (RELION4.0 is current default). Given that an unstable version is likely to be superseded by a stable release soon, it is inefficient to invest development efforts in it because repeating the optimization for each new software release requires considerable effort. Therefore, our basic policy is to support only the latest stable version as the default in GoToCloud. The final Step 3 GTC script obtains the URL of the NICE-DCV remote desktop environment on the head node of the pcluster instance constructed in Step 2. Accessing this link in a WEB browser, the user can display the OS desktop of the head node and start the analysis immediately (Supplementary Fig. S2).

When the user wishes to pause the analysis work and does not plan to use this GoToCloud platform instance for a while, it is recommended that the pcluster instance be deleted to stop billing for the relatively expensive head node and Lustre. For this additional step for pausing analysis work, the Step 4 GTC script is prepared. To resume the analysis work, the user can execute Steps 2 and 3 again. Upon completion of the project, the user can use the Step 5 GTC script to delete the associated AWS S3 bucket and EC2 key pair for permanent removal of the GoToCloud platform instance. Finally, the user should delete the Cloud9 IDE for the project from the AWS management console.

Tagging computational resources for cost calculation

Another important issue is the cost management. While on-demand usage is convenient, if users do not carefully control the use of computing resources, the cost can quickly expand. As it is partly the user's responsibility to use the necessary computing resources only when needed, careful attention is required for cost management. GoToCloud utilizes the tag feature of each AWS service for cost management. The tagging of all AWS services used in GoToCloud, including the head node, compute nodes, and storage of HPC clusters, is automatically performed by the GTC scripts in Step 1 and Step 2 based on the tag settings in Supplementary Table S4, eliminating the need for users to manually perform this task. By utilizing these tag settings, each research group can easily confirm the resource usage and its associated costs for each user and project.

Supporting multi-accounts and multi-regions

The Step 2 GTC script reduces the setup time of the software environment after generating pcluster instances by utilizing the shared EFS where related software packages are pre-installed. While sharing the analysis software environment between multiple AWS accounts brings various benefits as mentioned above, it is also necessary to ensure strong security at the AWS account level by building the shared EFS in the Amazon Virtual Private Cloud (VPC), which is a logically isolated virtual network that is not directly connected to the internet (Supplementary Fig. S7). Additionally, there are also numerous numbers of AWS Regions, each of which is a unit of AWS's cloud server data centre group and geographically separated from the other Regions. Therefore, the support for multi-regions was also addressed with GoToCloud along with the support for multi-accounts. For the details of the multi-account and multi-region support of the GoToCloud shared EFS using the VPC service, refer to 'section 1.2' in the Supplementary Information.

There are also multiple independent zones in each AWS region, known as availability zones, which indicate the physical location of the hardware. The resources available in each availability zone differ, particularly the set of available EC2 instance types, their quantities, and the availability of Lustre. Typically, to create a specific EC2 instance type, the user must select an AWS Region and an associated VPC, and then choose a subnet associated with one of the availability zones within the selected AWS Region. To eliminate the need for this selection process, the GoToCloud management group has pre-determined the most suitable availability zone for Cryo-EM SPA in each AWS Region and prepared a configuration file for CloudFormation, thereby reducing the user's burden.

Statistics and reproducibility

In this study, no statistical analyses were conducted as each data point in all graphs represents a single measurement. To effectively utilize the limited resources for our evaluation of GoToCloud, we decided to cover a wider range of parallel computing settings, specifically the types of RELION jobs, EC2 instance types, and the number of computing nodes, by sacrificing more robust reproducibility of each measurement. This approach allowed us to capture clear common trends in cost performance relative to the increase in computing resources, leading to a key finding from our benchmark tests: the saturation points of scalability, the minimum achievable cost, and the optimal parallel computing settings differ among Refine3D, Class3D, Class2D, and Polish jobs.

Reporting summary

Further information on research design is available in the Nature Portfolio Reporting Summary linked to this article.

Data availability

All data used in this paper are publicly available from EMPIAR and EMDB. For benchmark tests of the most important but time-consuming heavy processes in RELION, a dataset of native nitrite reductase (EMPIAR-10581[25]) with a small molecular weight of 110 kDa and C3 symmetry collected with 200 kV acceleration voltage was used to verify the effectiveness of GoToCloud. The resolution of the associated EMDB map is 2.85 Å (EMD-0731[34]). For assessment of the relationship between cost and resolution improvement, a dataset of streptavidin (EMPIAR-10641[30]) with a small molecular weight of ~53 kDa and D2 symmetry was used. The resolution of the associated EMDB map is 1.93 Å (EMD-30913)[31]. The detailed descriptions of sample handling, protein purification, Cryo-EM grid preparation, and data acquisition of these datasets can be found in previous publications[24]. All the relevant parameters of the data collection are summarized in Supplementary Table S1. All data and parameter settings needed to evaluate the conclusions in the paper are present in the paper, the Supplementary Information, and the Supplementary Data 1–3. All other data are available from the corresponding author (or other sources, as applicable) on reasonable request.

Code availability

All necessary scripts[23], including the GTC scripts and associated template files, for the construction of the GoToCloud platform and the shared EFS and shared S3 in the user's own AWS account are available at https://github.com/KEK-SBRC-CryoEM/gotocloud. The README file in the GitHub repository provides step-by-step instructions on how to use the GTC scripts for setting up the GoToCloud platform and creating the instances on the user's AWS account. It also includes links to websites where readers can find detailed versions of these procedures, comprehensive tutorials on using the GTC scripts, and installation instructions for the main software supported by the GoToCloud platform. The versions of relevant software used in this paper are RELION 4.0, RELION 3.1.2, UCSF Chimera version 1.14, AWS ParallelCluster v3.0.3 (Amazon Web Services, Inc.), GCC 9.3.0 compiler, GCC 9.4.0 compiler, Intel® oneAPI Base Toolkit 2022.1.2.146 (Intel Inc.), and Intel® oneAPI HPC Toolkit 2022.1.2.117 (Intel Inc.).

Received: 18 April 2024; Accepted: 7 October 2024;
Published online: 14 October 2024

References

1. Glaeser, R. M., Nogales, E. & Chiu, W. *Single-Particle Cryo-EM of Biological Macromolecules*. https://doi.org/10.1088/978-0-7503-3039-8 (IOP Publishing, 2021).
2. Van Drie, J. H. & Tong, L. Cryo-EM as a powerful tool for drug discovery. *Bioorganic Med. Chem. Lett.* **30**, 127524 (2020).
3. Merino, F. & Raunser, S. Electron Cryo-microscopy as a tool for structure-based drug development. *Angew. Chem. Int. Ed.* **56**, 2846–2860 (2017).
4. Renaud, J. P. et al. Cryo-EM in drug discovery: achievements, limitations and prospects. *Nat. Rev. Drug Discov.* **17**, 471–492 (2018).
5. Lees, J. A., Dias, J. M. & Han, S. Applications of Cryo-EM in small molecule and biologics drug design. *Biochem. Soc. Trans.* **49**, 2627–2638 (2021).
6. Wu, C., Huang, X., Cheng, J., Zhu, D. & Zhang, X. High-quality, high-throughput cryo-electron microscopy data collection via beam tilt and astigmatism-free beam-image shift. *J. Struct. Biol.* **208**, 0–1 (2019).
7. Weis, F. & Hagen, W. J. H. Combining high throughput and high quality for cryo-electron microscopy data collection. *Acta Crystallogr. Sect. D Struct. Biol.* **76**, 724–728 (2020).
8. Cheng, A. et al. High resolution single particle cryo-electron microscopy using beam-image shift. *J. Struct. Biol.* **204**, 270–275 (2018).

9. Wiley, K. et al. Astronomy in the cloud: using MapReduce for image co-addition. *Publ. Astron. Soc. Pacific* **123**, 366–380 (2011).
10. Jones, R. W. L. & Barberis, D. The evolution of the ATLAS computing model. *J. Phys. Conf. Ser.* **219**, 72037 (2010).
11. Hu, Y. S., Nan, X., Sengupta, P., Lippincott-Schwartz, J. & Cang, H. Accelerating 3B single-molecule super-resolution microscopy with cloud computing. *Nat. Methods* **10**, 96–97 (2013).
12. Krampis, K. et al. Cloud BioLinux: pre-configured and on-demand bioinformatics computing for the genomics community. *BMC Bioinformatics* **13**, 42 (2012).
13. Yazar, S., Gooden, G. E. C., Mackey, D. A. & Hewitt, A. W. Benchmarking undedicated cloud computing providers for analysis of genomic datasets. *PLoS One* **9**, e108490 (2014).
14. Mohammed, Y. et al. Cloud parallel processing of tandem mass spectrometry based proteomics data. *J. Proteome Res.* **11**, 5101–5108 (2012).
15. Trudgian, D. C. & Mirzaei, H. Cloud CPFP: a shotgun proteomics data analysis pipeline using cloud and high performance computing. *J. Proteome Res.* **11**, 6282–6290 (2012).
16. Cianfrocco, M. A. & Leschziner, A. E. Low cost, high performance processing of single particle cryo-electron microscopy data in the cloud. *Elife* **4**, e06664 (2015).
17. Cianfrocco, M. A., Lahiri, I., DiMaio, F. & Leschziner, A. E. cryoem-cloud-tools: a software platform to deploy and manage cryo-EM jobs in the cloud. *J. Struct. Biol.* **203**, 230–235 (2018).
18. Cuenca-Alba, J. et al. ScipionCloud: an integrative and interactive gateway for large scale cryo electron microscopy image processing on commercial and academic clouds. *J. Struct. Biol.* **200**, 20–27 (2017).
19. Baldwin, P. R. et al. Big data in cryoEM: automated collection, processing and accessibility of EM data. *Curr. Opin. Microbiol.* **43**, 1–8 (2018).
20. Punjani, A., Rubinstein, J. L., Fleet, D. J. & Brubaker, M. A. cryoSPARC: algorithms for rapid unsupervised cryo-EM structure determination. *Nat. Methods* **14**, 290–296 (2017).
21. Zivanov, J., Nakane, T. & Scheres, S. H. W. Estimation of high-order aberrations and anisotropic magnification from cryo-EM data sets in RELION-3.1. *IUCrJ* **7**, 253–267 (2020).
22. Kimanius, D., Dong, L., Sharov, G., Nakane, T. & Scheres, S. H. W. New tools for automated cryo-EM single-particle analysis in RELION-4.0. *Biochem. J.* **478**, 4169–4185 (2021).
23. Moriya, T., Yamamoto, M. & Yamada, Y. GoToCloud scripts and associated template files: fast stable release (version 01.00.00) [computer software]. Zenodo. https://doi.org/10.5281/zenodo.13842891 (2024).
24. Pettersen, E. F. et al. UCSF chimera—a visualization system for exploratory research and analysis. *J. Comput. Chem.* **25**, 1605–1612 (2004).
25. Adachi, N. et al. CryoEM map and model of nitrite reductase at pH 8.1. Electron microscopy public image archive (EMPIAR). https://doi.org/10.6019/EMPIAR-10581 (2020).
26. Adachi, N. et al. 2.85 and 2.99 Å resolution structures of 110 kDa nitrite reductase determined by 200 kV cryogenic electron microscopy. *J. Struct. Biol.* **213**, 107768 (2021).
27. Zivanov, J., Nakane, T. & Scheres, S. H. W. A Bayesian approach to beam-induced motion correction in cryo-EM single-particle analysis. *IUCrJ* **6**, 5–17 (2019).
28. Scheres, S. H. W. A Bayesian view on cryo-EM structure determination. *J. Mol. Biol.* **415**, 406–418 (2012).
29. Scheres, S. H. W. RELION: implementation of a Bayesian approach to cryo-EM structure determination. *J. Struct. Biol.* **180**, 519–530 (2012).
30. Hiraizumi, M., Yamashita, K., Nisihzawa, T., Kikkawa, M. & Nureki, O. 1.93 A cryo-EM structure of streptavidin. Electron Microscopy Public Image Archive (EMPIAR). https://doi.org/10.6019/EMPIAR-10641 (2021).
31. Hiraizumi, M., Yamashita, K., Nishizawa, T., Kikkawa, M. & Nureki, O. 1.93 A cryo-EM structure of streptavidin. Electron Microscopy Data Bank (EMDB). https://www.ebi.ac.uk/emdb/EMD-30913 (2021).
32. Wagner, T. et al. SPHIRE-crYOLO is a fast and accurate fully automated particle picker for cryo-EM. *Commun. Biol.* **2**, 1–13 (2019).
33. Rohou, A. & Grigorieff, N. CTFFIND4: fast and accurate defocus estimation from electron micrographs. *J. Struct. Biol.* **192**, 216–221 (2015).
34. Adachi, N. et al. CryoEM map and model of nitrite reductase at pH 8.1. Electron Microscopy Data Bank (EMDB). https://www.ebi.ac.uk/emdb/EMD-0731 (2021).

Acknowledgements

The authors thank Koji Tashiro, Daisuke Miyamoto, Takeshi Sakurada, Hidenori Koizumi, Jorge Lanzarotti, and Tomoyoshi Ootsubo (Amazon Web Services Japan G.K.) for AWS Cloud-related technical advice and general support; Stephen Litster (Amazon Web Services, Inc.) and Brendan Bouffler (Amazon Web Services EMEA SARL, UK Branch) for providing AWS credits for our benchmark tests; Michael J Mcmanus, Jason Do, Katsumi Yazawa, and Yuka Shimizu (Intel Corporation) for sharing their benchmark results of CPU-Optimized RELION and related technical advice; Masahiro Hiraizumi, Keitaro Yamashita, and Tomohiro Nishizawa for sharing the streptavidin dataset and its information; Naruhiko Adachi, Masato Kawasaki, Akihito Ikeda, and Satomi Inaba for helpful discussion of the Cryo-EM SPA workflow; and Rieko Sukegawa and Chiho Masuda for assistance with the Cryo-EM study. This study was supported by the Platform Project for Supporting Drug Discovery and Life Science Research [Basis for Supporting Innovative Drug Discovery and Life Science Research (BINDS)] from AMED under grant number JP23ama121001 (T.S.), and JSPS KAKENHI grant numbers JP20K15735 and JP23H02427 (T.M.).

Author contributions

T.M., Y.Y., and T.S. conceived and designed the experiments. T.M., Y.Y., and M.Y. established the GoToCloud platform for data analysis of the Cryo-EM SPA on the AWS Cloud service. T.M. and M.Y. processed the Cryo-EM data. T.M., Y.Y., M.Y., and T.S. prepared the manuscript.

Competing interests

Y.Y. and T.M. declare the following competing interests: they are the inventors of a pending patent application (Japanese patent application 2022-211645 filed on December 28th, 2022, by the High Energy Accelerator Research Organization) related to the technology described in this paper. The other authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s42003-024-07031-6.

Correspondence and requests for materials should be addressed to Toshio Moriya or Toshiya Senda.

Peer review information *Communications Biology* thanks Michael Cianfrocco, Billy K. Poon, and the other, anonymous, reviewer for their contribution to the peer review of this work. Primary handling editors: Aylin Bircan and Ophelia Bu. A peer review file is available.