Neural Networks 144 (2021) 478–495

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

2021 Special Issue on AI and Brain Science: AI-powered Brain Science

The whole brain architecture approach: Accelerating the development of artificial general intelligence by referring to the brain

Hiroshi Yamakawa [a,b,c,*]

[a] The Whole Brain Architecture Initiative, Nishikoiwa 2-19-21, Edogawa-ku, Tokyo 133-0057, Japan
[b] The University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo 113-0033, Japan
[c] RIKEN, 6-2-3, Furuedai, Suita, Osaka 565-0874, Japan

ARTICLE INFO

Article history:
Available online 14 September 2021

Keywords:
Brain reference architecture
Structure-constrained interface decomposition method
Brain information flow
Hypothetical component diagram
Brain-inspired artificial general intelligence
Whole-brain architecture

ABSTRACT

The vastness of the design space that is created by the combination of numerous computational mechanisms, including machine learning, is an obstacle to creating artificial general intelligence (AGI). Brain-inspired AGI development; that is, the reduction of the design space to resemble a biological brain more closely, is a promising approach for solving this problem. However, it is difficult for an individual to design a software program that corresponds to the entire brain as the neuroscientific data that are required to understand the architecture of the brain are extensive and complicated. The whole-brain architecture approach divides the brain-inspired AGI development process into the task of designing the brain reference architecture (BRA), which provides the flow of information and a diagram of the corresponding components, and the task of developing each component using the BRA. This is known as BRA-driven development. Another difficulty lies in the extraction of the operating principles that are necessary for reproducing the cognitive–behavioral function of the brain from neuroscience data. Therefore, this study proposes structure-constrained interface decomposition (SCID), which is a hypothesis-building method for creating a hypothetical component diagram that is consistent with neuroscientific findings. The application of this approach has been initiated for constructing various regions of the brain. In the future, we will examine methods for evaluating the biological plausibility of brain-inspired software. This evaluation will also be used to prioritize different computational mechanisms, which should be integrated and associated with the same regions of the brain.

1. Introduction

Artificial general intelligence (AGI), the development of which has been a major goal in advanced artificial intelligence (AI) research in recent years, involves demonstrating the extensive general intelligence that is possessed by humans within a computational system (Adams et al., 2012; Goertzel, 2014). An essential ability of AGI could be solving various problems, including those on unknown issues, by flexibly combining knowledge that is gained from experience. However, methods for developing AGI remain unclear. Many AI researchers believe that the development of deep learning (LeCun, Bengio, & Hinton, 2015) serves as a launch pad for this goal. According to these scholars, this goal can be realized by combining various computational mechanisms, including machine learning, which is a method that enables a machine to learn knowledge from experience. Several attempts have been made to create a unified theory and principle of intelligence (Domingos, 2015; Friston, 2010; Hafner et al., 2020). However, no single theory exists on which the entire sphere of intelligence can be built. Thus far, the development of AGI has progressed by the repetitive tuning of various limited issues. However, such an approach makes it difficult to design AGI with flexible problem-solving abilities that would enable unknown issues to be solved. The construction of an AGI that possesses the full extent of human abilities would require an extremely large design space owing to the combination of computational mechanisms. Although this design space could also be explored mechanically (Clune, 2019), at present, it is difficult to secure the required computational complexity.

Brain intelligence is associated with a certain degree of versatility. The development of an AGI that is comparable to human intelligence may be accelerated by narrowing the design space by referring to the architecture of the cognitive and behavioral functions in the brain (Petersen & Sporns, 2015). That is, even if the scope of AGI realized by machines (the machine kingdom) does not need to be bound by biological constraints (Hernández-Orallo, 2017), the development of a brain-like architecture could be a significant milestone in AGI development (Goertzel, Lian, Arel, de Garis, & Chen, 2010; Hassabis, Kumaran, Summerfield, & Botvinick, 2017).

* Corresponding author.
E-mail address: ymkw@wba-initiative.org.

https://doi.org/10.1016/j.neunet.2021.09.004

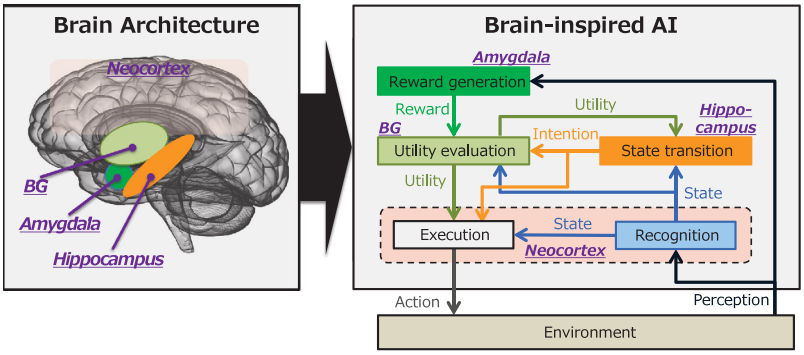

Fig. 1. Basic scheme of WBA approach. This is a revised version of a simplified diagram of the basic concept of the WBA approach, which has been gradually formed since 2014. The left side of the figure presents major examples of large organs in the brain, including the neocortex, basal ganglia , hippocampus, and amygdala. An additional brain architecture is formed by the connections among these organs (not shown). On the right side of the figure, the computational modules, including those that utilize machine learning, are placed and connected with reference to the brain architecture. This scheme forms the basis for the construction of an AI software system that can operate while interacting with the environment through the body.

Based on this technological background, we at the Whole Brain Architecture Initiative (WBAI) have been advocating for a developmental method known as the whole-brain architecture (WBA) approach since 2015. We define the basic idea of this AGI development approach as the creation of "human-like artificial general intelligence by learning from the architecture of the entire brain" (Arakawa & Yamakawa, 2016; Yamakawa, Osawa, & Matsuo, 2016).

The premise of this approach is known as the Central WBA Hypothesis, which is expressed as follows: "The brain combines modules, each of which can be modeled with a machine learning algorithm, to attain its functionalities, so that the combination of machine learning modules in the same manner as the brain will enable us to construct a generally intelligent machine with human-level or super-human cognitive capabilities".

According to these assumptions, the aim of the WBA approach is to construct a brain-inspired AGI based on the following basic concepts: As illustrated in Fig. 1, each brain organ is implemented as a calculation module, including those that utilize machine learning, and these are integrated based on the brain architecture. The brain organs depicted in the figure are fairly coarse, but in reality, they are associated with the brain in units of finer-grained computational modules (see Section 2.1).

It is not realistic to construct brain-inspired AGI software by directly referring to the neuroscientific findings in academic papers and data. This is because the functions of the brain are diverse and vast neuroscientific findings regarding these functions are available. Furthermore, the number of people who thoroughly understand neuroscience and can develop software is limited, as this field involves intensive training.

To address this problem, the WBAI has standardized the information corresponding to the requirements for developing brain-inspired software in the form of brain reference architecture (BRA) data. The BRA design and implementation methods are presented in Fig. 2 (Sasaki, Yamakawa, & Arakawa, 2020), which we refer to as BRA-driven development.

The BRA is the reference architecture for the neural circuits of the brain (see Section 2), which basically consists of a description of the brain information flow (BIF) and one or more associated hypothetical component diagrams (HCDs). The BIF describes the anatomy at the mesoscopic level as a directed graph connecting nodes that represent the local neural circuits (see Section 2.2). The HCD is a directed graph that describes the dependencies that are formed by the components of the computational functions. This directed graph is a hypothesis that is designed to be included in a directed graph described by a BIF (see Section 2.4). The neural behavior and process (NBP) provides a description of the

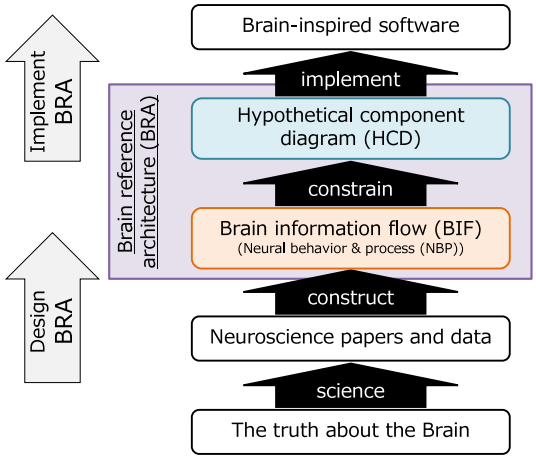

Fig. 2. BRA-driven development, which consists of developing brain-inspired software using the BRA and designing the BRA based on neuroscientific knowledge (studies and data). The BRA consists of the BIF and HCD.

neural activity in the region of interest (ROI) that often includes useful hints for the HCD design, although its role in BRA-driven development is somewhat supplementary (see Section 2.3).

In general, even if the target BIF is specified, the functional hypothesis thereof cannot be uniquely determined. This is because different tasks and capabilities are used as the starting point for designing the function, the anatomical granularities that are addressed differ, and the knowledge that is required to specify the function is insufficient. To deal with these factors, the BRA format enables the data of multiple HCDs to be described for unique BIF data. Nevertheless, the inclusion of HCDs with poor biological plausibility in BRA data should be avoided. Therefore, the HCDs themselves should only be formally registered if they have an appropriate function and remain consistent with the BIF (see Section 3.3).

Thus, it will be easier to compare and evaluate the validity of multiple hypotheses of computational functions if these hypotheses can be described using the BRA data description format, which explains the computational functions in a standardized manner that is grounded in anatomical structures. Data that are prepared in this manner can facilitate the derivation of a highly general hypothesis that integrates multiple hypotheses. Furthermore, by making it easier to examine the consistency of hypotheses with other hypotheses that are assumed for the surrounding neural circuits, it will be easier to examine the validity of the hypotheses

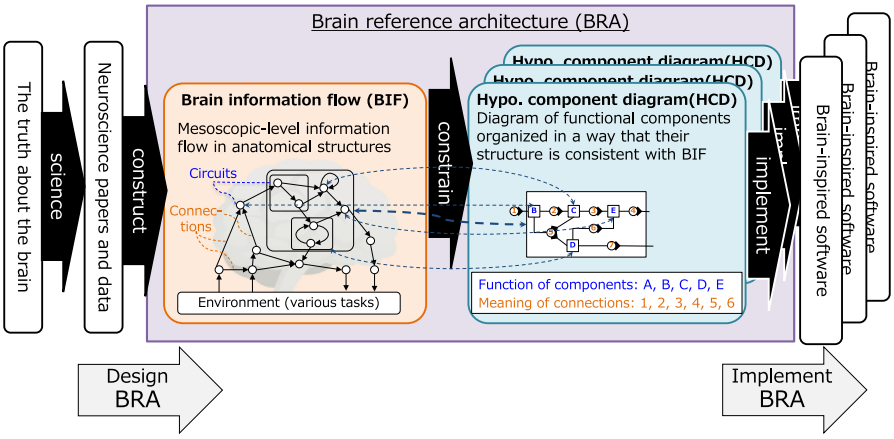

Fig. 3. BRA. The BRA consists of the BIF and HCD. The BIF provides the mesoscopic-level information flow in anatomical structures. The HCD is a diagram that organizes the functions in a manner that is consistent with the anatomy of a given circuit. Multiple HCDs can be used for a single circuit in the BIF. Every software development project is essentially based on a specific HCD.

from a broader perspective, which is difficult to achieve when focusing on a narrow range.

Therefore, even a developer with limited knowledge of neuroscience can implement the software if the BRA is provided, and the HCD that is contained in the BRA is regarded as a requirement.

The development of brain-inspired software requires HCDs that cover a wide range of brain regions according to its purpose.[1] However, the creation of such HCDs is not an easy task. Even with the recent rapid developments in neuroscience, operating principles have been described for only several regions of the brain.

To address this issue, structure-constrained interface decomposition (SCID) has been developed as a research method to elucidate the functional mechanisms by focusing on brain circuits. The SCID method attempts to obtain the functional mechanisms by decomposing the entire function of a particular brain region to be consistent with the anatomical structure at the mesoscopic level, as described in detail in Section 3.2. A fairly wide range of brain mesoscopic anatomical structures has been established in current neuroscience. Therefore, by using the SCID method, we can create BRAs of a relatively wide range of brain regions, while supplementing the functional mechanisms that are not yet clear at present.

As BRA-driven developments progress, their deliverables tend to move away from the reality of the brain, which is a problem because the WBA approach aims to explore AGI within a design space that is similar to that of the brain. To overcome this problem, it is necessary to continue to evaluate how effectively the implemented brain-inspired software reproduces the truth of the brain (the parts relating to the level of cognitive behavior) as perceived by neuroscience. The evaluation of such biological plausibility is carried out from two viewpoints. The first evaluation is adequate for BRA to assess whether it is consistent with the existing neuroscience findings. The other is the evaluation of the fidelity, or whether the software is built according to the BRA (Yamakawa, 2020c; Yamakawa, Arakawa, & Takahashi, 2020).

System integration has become an important issue in the later stages of AGI development. Software development is usually performed to realize a certain task, but various implementations will inevitably be created in the process. Such disparate implementations that are performed in the first half of the AGI development are integrated into the second half, and only then can the intelligence be generalized. At this stage, the feature of BRA-driven development comes into play, whereby each implementation corresponds to a common BIF. That is, the components to be integrated between different implementations can be specified via the BIF. This enables the integration of the entire system to be decomposed into the code integration of each component. Thus, the system integration can be performed more efficiently. We refer to this process as brain-inspired refactoring (see Section 4.3).

The remainder of this paper is organized as follows: Section 2 delves into the BRA and discusses the description level at which the brain-inspired AGI should learn about the brain, the BIF format, which is an element for describing the BRA, and the HCD. Furthermore, we provide an overview of BRA-driven development. Section 3 presents the collection of neuroscientific findings relating to the BRA design, SCID, and evaluation of the BRA validity. The stub-driven and integration development using the BRA, as well as methods for evaluating the fidelity of software from the perspective of the BRA in the future, are discussed in Section 4. Discussions are presented in Section 5, and finally, Section 6 concludes the paper.

2. Brain Reference Architecture (BRA)

The BRA is the reference architecture for software that realizes cognitive and behavioral functions in a brain-like manner. The architecture primarily consists of the mesoscopic-level anatomical data of the brain and the data of one or more functional mechanisms that are consistent with that knowledge. In particular, as illustrated in Fig. 3, the data are a combination of the BIF (see Section 2.2) and HCD (see Section 2.4) (Sasaki et al., 2020).

The current WBA approach is based on BRA-driven development. This development consists of the design of the BRA, the evaluation thereof based on neuroscience findings, and the implementation and evaluation of the software with reference to the HCD within the BRA. After describing the BRA at the mesoscopic level, we discuss the BIF as a component of the BRA and HCD, which are designed to be consistent with the BIF. Moreover, we explain BRA-driven development.

2.1. Mesoscopic levels to be referenced in BRA

Which level of granularity of the brain should be described using BRA data? The central nervous system has an anatomical

[1] If the development target is brain-inspired AGI, intelligence similar to that of humans must be comprehensively constructed, so the area to be covered is almost the entire brain.

hierarchical network structure. Therefore, it is natural to realize software that refers to the brain as a network of several functional components. By standardizing the granularity of the brain-referenced components, it will be easier to refer to these during software development and to integrate multiple BRA data.

In the use of the BRA as design data, the simplest concept of unifying the granularity of the description in each neuron is not realistic owing to the following points. First, the design of too many parts should be avoided. For example, considering that a car is composed of tens of thousands of parts and an airplane is composed of millions of parts, it is not realistic to draw a blueprint with more than 10 billion human neurons. Second, the learning factors following maturation can be ignored. This means that the detailed connections between neurons, which vary depending on individual experience, should not be designed, but rather, should be tuned by machine learning.

Therefore, the descriptive granularity of BRA data should be determined at an intermediate level (the mesoscopic level) in a hierarchy of the neural circuits in the brain. However, it is obvious that a mesoscopic level of granularity that relies on physical measures to determine the voxel size of fMRI measurements is inappropriate.

2.1.1. Uniform circuits: Arguments of software components in brain

The mesoscopic-level granularity of the brain circuitry that should be referenced in the BRA data is clarified in terms of the smallest elements to be described in the software design.

As mentioned previously, brain-inspired software is composed of a network of components. Therefore, the contents that should be described in the BRA data are the external functional specifications and interfaces of each component as well as the connections among the components. The arguments that are described in the interface of the component are among the smallest elements included in these design elements. For example, in a reinforcement learning program, the interface of a component may contain arguments such as states, actions, and rewards that are represented by one- or multi-dimensional vectors. Therefore, the identification of the entities in the brain corresponding to the arguments leads to the determination of the description granularity of the BRA data.

The existence of a physical entity corresponding to the arguments should be assumed to use the brain as a reference architecture for software. As these arguments with specific meanings are variables that change continually, they are represented as the activity of neurons in the brain. Furthermore, because the representation of information in the brain is generally redundant, a group of neurons that encode similar meanings plays the role of the argument; therefore, the group of neurons in the brain corresponding to the argument is defined as the following uniform circuit.

Uniform circuit:

A uniform circuit is a group of neurons in the brain that can be regarded as encoding the same type of meaning functionally.

2.1.2. Uniform circuit as a group of neurons composed of a specific cell type

In the nervous system, synaptic specificity is a property that controls the combination of cells in which synaptic connections are established (de Wit & Ghosh, 2016; Williams, de Wit, & Ghosh, 2010). This is the property whereby an axon projecting from a particular cell type selectively forms synaptic connections at the receiving end to a particular laminar, cell type, and location within the cell. Synaptic specificity enables the cells at the receiving end to identify neural groups of the same cell type at the sending end that they wish to use for processing, as well as to distinguish among projections from different cell types.

Even interregional axon projections can be directed to precise target cells for each cell type by means of a process known as axon guidance. In this process, the elongation direction of the growth cone at the tip of the axon is controlled through different responses of each cell type to surrounding guidance molecules. This process also contributes to the formation of topographic maps, which are projections that preserve the two-dimensional spatial relationships between different regions, such as retinotopy (Triplett et al., 2009). A mechanism that is similar to transferring two-dimensional array arguments between software components can be constructed using this process.

In light of the above discussion, it is reasonable to consider a group of neurons that are composed of a specific cell type in the source domain as an argument for a software component. Therefore, we make the following assumptions regarding the neuroscientific foundation of the neurons that constitute the uniform circuits.

Cell type-based uniform circuit hypothesis:

A uniform circuit is formed by a group of neurons that are composed of a specific cell type within a particular brain region.

According to this hypothesis, the neuronal group within a particular brain region that encodes the information to provide the argument of the software is composed of certain unique cell types. That is, the uniform circuit, which is the minimum granularity used to describe the BRA, can be set at the mesoscopic level as a group of neurons that are composed of a specific cell type (see Fig. 4A).

Furthermore, Bohland et al. (2009) noted that mesoscopic-level architecture, which is a unit of cell groups classified by the same cell types that are localized in a certain brain region, has a significant impact on cognitive behavioral functions. Thus, it is reasonable to consider the correspondents of the arguments exchanged in the architecture as neural groups consisting of specific cell types.

> Significantly more invariance can be expected at a mesoscopic level where co-localized groups of neurons, perhaps of the same type or sharing common organizational features, are considered together as a unit, and projection patterns from these neuronal groups are studied over macroscopic distances. This level of connectivity is well-suited to aid our understanding of specific mental functions. —(Bohland et al., 2009).

2.1.3. Diversity of uniform circuit description

Diverse aspects of cell types exist, such as the physiological and morphological features, gene expression, anatomical location, and projection patterns, with gradations in the similarity of each aspect (Mitra, 2014). However, our knowledge of mammalian synaptic specificity remains inadequate; therefore, it is still difficult to select appropriate features that contribute to the classification of uniform circuits from the wide variety of feature axes of cell types.

A pragmatic approach is to select the cell type features that classify uniform circuits so that they can distinguish the arguments of components that are required for the functional design of brain-inspired software. It is inevitable that the granularity of the description of uniform circuits will vary depending on the target task when designing the BRA. Therefore, at present, it is difficult to avoid the appearance of uniform circuits with different overlapping granularities completely in the description of BRA data.

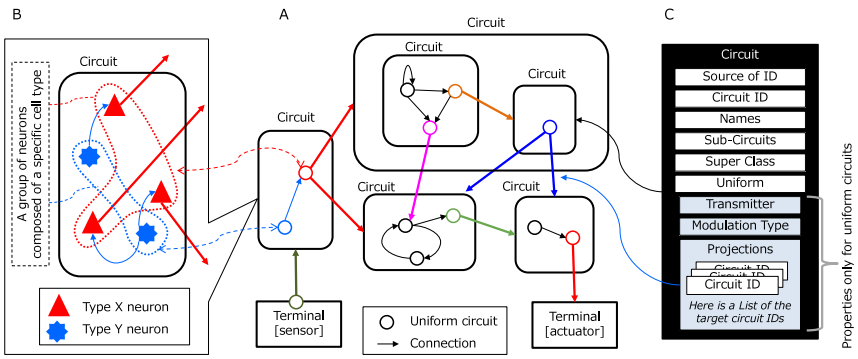

Fig. 4. BIF. The BIF describes the flow of information in anatomical structures at the mesoscopic level of the brain. A: A BIF is a graph consisting of circuits and the connections among them. Each circuit is any organ or region of the brain and the assemblies to which they are connected. The starting point of a connection is a uniform circuit that is functionally considered to encode the same type of meaning. B: The leftmost circuit in A is enlarged and the two types of neurons inside it are depicted schematically. Reflecting synaptic specificity, each uniform circuit is assumed to be formed by a group of neurons composed of a specific cell type within a particular brain region. C: Attributes describing each circuit in the BIF data. Certain attributes are specific to uniform circuits, the most representative of which is the projection attribute, which is a list of circuit IDs of the projection targets.

Table 1
Attributes of BIF and NBP data.

Attribute	Description	Values
Source of ID	Source of circuit ID	*Ontology ID*/*Reference ID*/"collection"/"makeshift"
Circuit ID	Identifier of circuit	*string*
Names	Circuit labels	*string*
Sub-circuits	List of circuits to include	List of *circuit IDs*
Super-class	Upper class	List of *circuit IDs*
Uniform	Whether uniform circuit	True/false
Transmitter	Type of neurotransmitter	Glutamate/Dopamine/Acetylcholine/GABA
Modulation type	Functional form of neurotransmission	Excitatory/Inhibitory/Modulatory
Size	Number of neurons	*text* [RID]
Projections	ID of circuit to which axon projects	List of *circuit IDs* [RID]
Interpretation	Description of physiological phenomena, etc.	*text* [RID]
Physiological data	Neural activity data	Index to data

[RID]: An attribute that requires a *reference ID*.

2.2. Brain information flow (BIF)

The BIF describes the anatomical structure of the entire brain at the mesoscopic level (Arakawa & Yamakawa, 2020) (see Fig. 4. As such, it is not intended for specific tasks in the environment. The BIF is a graph, the basic structure of which consists of a node known as a "circuit" and a directed link known as a "connection". The smallest unit of the graph is the uniform circuit defined in Section 2.1, which also serves as the starting point for a connection. Moreover, a circuit is a graph that contains multiple uniform circuits, and multiple circuits may have overlapping portions.

2.2.1. Circuits (nodes)

A circuit is a component that becomes a node in the graph structure of the BIF. A uniform circuit is a group of neurons in the brain that can be regarded as functionally encoding the same kind of meaning. Subsequently, each uniform circuit is the lower limit of the BIF granularity and can provide a starting point for a connection. In general, a circuit may be any sub-circuit in the brain. This may indicate areas such as the entire visual cortex or only V1 (i.e., the primary visual cortex), or it may correspond to the neocortex–basal ganglia loop.

As indicated in Table 1, the attributes possessed by all circuits include the source of ID, circuit ID, names, sub-circuits, super-class, and uniform. Furthermore, the unique attributes of the uniform circuit include the transmitter, modulation type, size, and projections.

The circuits and connections are discussed in the following.

2.2.2. Connections (links)

Connections correspond to bundles of axons that are responsible for signal transmission between circuits in the brain, which are represented by links in Fig. 4A. The connections are described by a list of projection attributes on a uniform circuit. The number of axons for each species can be added to the description for each projection attribute.

2.3. Neural Behavior and Process (NBP)

The NBP describes the knowledge of neuroscience of dynamic physical phenomena. The main objects described in this case are the behaviors of neural activities and their combined processes that occur in the ROIs in response to the task being performed. Such dynamic findings in neuroscience are often useful for examining functions in the HCD, which include "interpretation" and "physiological data".

2.4. Hypothetical component diagram (HCD)

The component diagram of the unified modeling language (Ambler, 2004) is used to model and explain the structure of any complex object-oriented software. This diagram depicts the structural aspects of the functional mechanism of software as a network using socket labels to indicate the components that are responsible for the computing functions, as well as the interfaces of the call relationships between those components (see right image in Fig. 5). The diagram is used to visualize, specify,

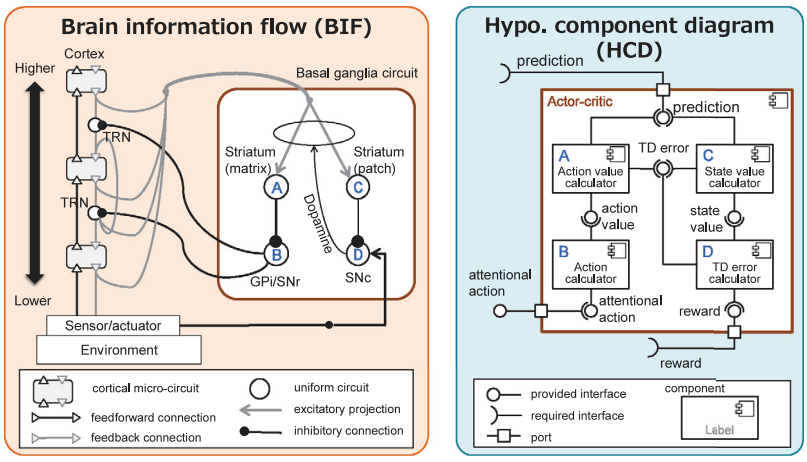

Fig. 5. Example of a BRA description that associates the BIF for the basal ganglia with the HCD for actor–critic reinforcement learning. The blue letters A, B, C, and D represent the uniform circuits in the ROI of the BIF (left panel). The corresponding components in the HCD (right panel) are marked with the same letters, and the HCD components are marked with their functions. The meaning of the signal to be transmitted is indicated by the interface between the components. In the BIF, TRN indicates a thalamic relay neuron. This diagram was adapted from the diagram in Takahashi, Schoenbaum, and Niv (2008) and Yamakawa (2020a).

document, and build an executable system by forward or reverse engineering.

The HCD that constitutes the BRA is a type of component diagram. This diagram assigns components that match the function of the brain ROI with an anatomical structure at the mesoscopic level; however, it is hypothetical as there is no guarantee that it is consistent with the truth of the brain function. The assignment is performed using the SCID method, which is discussed later. Each component that constitutes an HCD is a module that encapsulates a set of related functions (or data)[2] and corresponds to the behavior and structure of specific brain organs and regions.

As a typical example of a BRA, the association between the BIF and HCD is depicted in Fig. 5, which shows the well-known example (Takahashi et al., 2008) of the actor–critic reinforcement learning function of the basal ganglia. In the left diagram depicting the basal ganglia loop, the basal ganglia circuit is the ROI. The corresponding HCD that decomposes the actor–critic reinforcement learning function is presented on the right. The uniform circuit, named striatum (matrix) and indicated by the letter A in the BIF diagram, corresponds to the action value calculator component indicated by the letter A in the HCD. Similarly, the uniform circuits in the BIF correspond to the components in the HCD, as indicated by the other letters (B, C, and D). The following is an example of mapping the links between the two diagrams. The signal path (labeled dopamine) that is output from the SNc, indicated by the letter D in the BIF, is mapped to the signal path (labeled TD error) that is output from the TD error calculator component, indicated by the letter D in the HCD. Note that, in this example, the circuits indicated by A, B, C, and D in the BIF are all uniform circuits. Therefore, the label names assigned to the respective components in the HCD correspond to the label names of the arguments that they provide.

In this manner, the availability of an HCD, which displays the structural aspect of the functional mechanisms, increases the likelihood that even developers without profound expertise in neuroscience will be capable of implementing software that is closer to the truth of the brain. Network machine learning systems that are frequently used in current AI research (e.g., artificial neural networks and Bayesian networks) are compatible with development owing to component diagrams.

[2] The term "component" is also used in software engineering to refer to software packages, web services, web resources, and similar entities; however, in this paper, we use it to refer to a module that encapsulates a set of related functions (or data).

2.5. Prototype of BRA database

The brain is a fairly closely linked system. Therefore, the accumulation of standard neuroscientific findings relating to cognitive behavior will not only optimize the development of brain-inspired software, but will also aid in comprehensively grasping mesoscopic findings in the entire brain.

In this regard, the WBA approach examines databases to improve reusability by integrating the constructed BRA. In this study, a prototype of the BIF database was constructed using Semantic MediaWiki.

The data flow proceeded as follows: First, one of the authors with expertise in neuroscience reviewed the academic papers and compiled the relevant data in a spreadsheet. Subsequently, the data were registered in a database using a conversion tool. Thereafter, when the developers implemented the brain-inspired software, a tool prototype was created, which not only could browse the data directly, but could also visualize the BRA data as a graph in the ROIs.

Such activity can also be positioned as part of the field of neuroinformatics (Amari et al., 2002; Pradeep, Knight, & Gurumoorthy, 2013), in which data- and knowledge bases are developed for neuroscience. At present, experimental data on anatomical structures (Kuan et al., 2015) and physiological phenomena (Poldrack & Gorgolewski, 2017) are being vigorously registered in this field. However, no progress has been made in the accumulation of data for designing cognitive and behavioral functions, such as BRA.

2.6. BRA-driven development

BRA-driven development is a developmental approach that constructs brain-inspired AGI through the following processes using a standardized BRA (see Fig. 6).

- Design of BRA: The design of the BIF by collecting and organizing neuroscientific findings. Furthermore, an HCD is created using the SCID method (discussed later).
- Implementation of BRA: The implementation of brain-inspired software by referring to the HCD in the BRA.

In this manner, the developer can develop brain-inspired software guided by the HCDs in the BRA and can compensate for the lack of expertise in both neuroscience and software engineering.

In the following, we provide an overview of the three activities involved in the BRA design. The first activity is laying the foundation for accumulating BRAs that are useful for brain-inspired

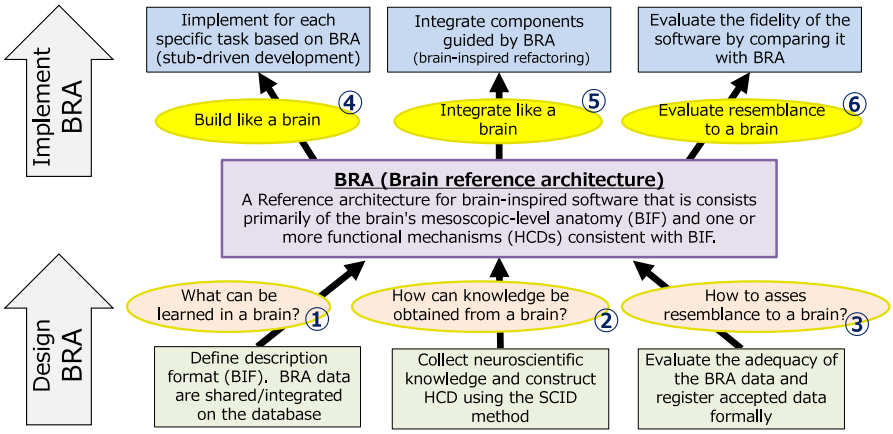

Fig. 6. Activities of BRA-driven development. The major activities of BRA-driven development include three types for designing the BRA and three types for using the BRA.

software. Specifically, this involves determining the description format (BIF; Section 2.2) and examining databases for the integrated sharing of BRAs (Section 2.5). (① What can be learned from the brain?) The next step is to acquire and formulate the knowledge that is necessary for the BRA. Specifically, knowledge regarding the anatomical structures and psychological phenomena in certain regions of the brain is collected and amassed in the form of the BIF. These BIF data are used to construct the HCDs through the SCID method. (② How can knowledge be obtained from the brain?) The third activity involves evaluating the adequacy of the BRA data. In this activity, the review criteria are determined and a judgment is made on whether the created BRA data satisfy the necessary requirements as a reference model for brain-inspired software. (③ How to assess the resemblance to the brain.)

Furthermore, we present an overview of the three activities that are associated with BRA utilization. First, in the development of brain-inspired software, the HCD that is associated with a specific task in the BRA is implemented as a requirement. (④ Build software similar to a brain.) In the future, we plan to carry out integration development, whereby components in separately developed programs are associated with one another based on the BRA and integrated. (⑤ Integrate disparate functions as the brain does.) Furthermore, to estimate how effectively the implemented software represents the brain, the fidelity (biological plausibility) is evaluated by comparing the BRA and program. (⑥ Evaluate the resemblance to the brain.)

As mentioned previously, it is necessary to integrate multiple computational mechanisms that correspond to the same brain regions and have been created according to, for example, the diversity of tasks, in each BRA-based development project to complete the brain-inspired AGI. Therefore, we believe that the entire development based on the WBA approach in the near future will proceed in parallel or iteratively with BRA-based and integration developments. Thus, the fidelity evaluation of the software will prevent the developmental results from veering away from the brain architecture.

3. Design of BRA

The design of the BRA is described in this section. Among the three activities related to the BRA design, the first one, namely "① What can be learned from the brain?", has been described in Section 2. Regarding "② How can knowledge be obtained from the brain?", the collection of anatomical findings in the neuroscience field and the SCID method for HCD construction are explained. Moreover, the assessment of appropriateness is discussed in the context of "③ How to assess the resemblance to the brain". The description format of the BRA data thus produced is summarized in Section 3.4.

3.1. Neuroscientific findings available for BIF and NBP creation

We discuss the process of acquiring the information relating to the anatomical structure that is required to describe the BIF (see Table 1). The main requirement is information for building directed graphs with circuits as nodes. Therefore, it is ideal to acquire information on uniform circuits of the entire brain and the connections among these circuits. The current state of neuroscience remains far from acquiring ideal information. In this regard, if necessary, circuits that are larger than the uniform circuit can be defined, and a BIF graph will be constructed among these circuits.

The information to be acquired for each uniform circuit includes the brain region labels (circuit IDs), animal species, neurotransmitters, excitatory and inhibitory modes, cell count, and information sources (references). The information to be acquired for the connections includes the input circuit, output circuit, animal species, size (number of axons), neurotransmitters, and sources (references). The orientation of the hierarchy between areas (including feedforward/feedback) is required for the neocortex.

Furthermore, the data that are described in the BIF are used to implement the software using an artificial neural network. Thus, it is ideal to know the number of neurons in a circuit and the approximate connection sizes (number of axons).

It is clear that a BRA that is used to construct human-like intelligence should be based on the structure of the human brain. However, it may be possible to streamline the construction of the BIF by referring to the findings in other animals, particularly rodents. Therefore, in reality, the BIF mainly uses human data for the neocortex, which is unique to humans; however, for other brain regions, several references based on non-human primates and rodents are incorporated (Negishi, Hayami, Tamura, Mizutani, & Yamakawa, 2019). Thus, although it exhibits similarity to humans overall, the BIF appears to contain chimeric data that combine mesoscopic-level anatomical findings from multiple mammals.

3.1.1. Information sources

The main information sources for constructing a BIF are the data regarding anatomical structures (such as connectomes) and the related literature.

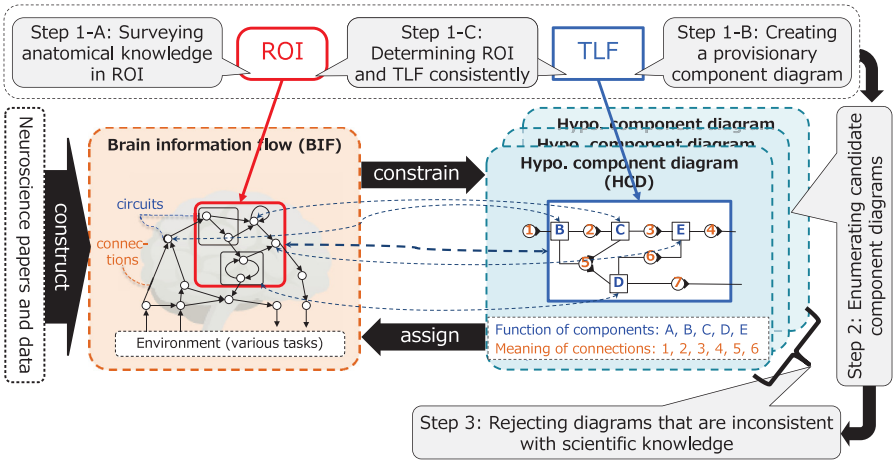

Fig. 7. Procedure of SCID method. The SCID method is a technique for obtaining the HCDs that are required for software development. The method uses a three-step process to decompose the TLFs of a specific brain region (ROI) into components under mesoscopic-level anatomical constraints. In step 1, three exploratory tasks (1-A, 1-B, and 1-C) are performed in parallel, as appropriate. Thereafter, we proceed to steps 2 and 3.

3.1.2. Region labels (circuit IDs)

In principle, the Allen Developing Human Brain Atlas ontology, which is an Allen Brain Reference Atlas (https://atlas.brain-map.org/), is used as a region label (circuit ID); if necessary, a label with a level that roughly corresponds to the granularity of the uniform circuit is added.

3.1.3. Number of neurons (size)

The number of neurons in each region of the mouse brain is stored in the Blue Brain Cell Atlas (Erö, Gewaltig, Keller, & Markram, 2018). These regions are defined using the Allen Mouse Brain Reference Atlas (Kuan et al., 2015). However, no comprehensive data on the number of neurons in humans are available at present.

3.1.4. Connections

It is desirable to gather information on the presence of the connections among circuits, their directions, and the approximate number of projection axons for all combinations of areas.

Although this is not necessarily an exhaustive brain region at all, the projection ratio from one particular area to another can be estimated using the Allen Mouse Brain Connectivity Atlas (Oh et al., 2014). As mentioned previously, because the number of neurons in an area can be obtained for mice, the number of axons to be projected can be estimated by multiplying the projection ratio by the number of neurons.

The Multilevel Human Brain Atlas by EBRAIN[3] can be used to obtain human data, including the hierarchical relationships (feedforward/feedback), for the entire neocortex.

3.1.5. Neurotransmitters

Although data on the distribution of neurotransmitters throughout the brain are currently available from Drosophilidae studies (Meissner et al., 2019), it appears that no data for mammals exist. However, similar anatomical structures appear frequently in each brain region that is involved in higher intelligence processing, such as the neocortex, thalamus, basal ganglia, hippocampus, and cerebellum. Moreover, as the neurotransmitters in these sites have been studied in detail, the lack of data does not pose a major problem in the BIF construction. Nevertheless, data on the subcortical brain regions are required.

3.1.6. Interpretation

Physiological phenomena can be described; for example, "burst firing". Furthermore, a highly reliable functional interpretation of these phenomena can be provided (e.g., grid cells).

3.1.7. Physiological data

These data provide a reference to the neural activity data in the ROI, such as URLs and drawings in papers.

3.2. Structure-constrained interface decomposition (SCID) method

The SCID method involves consistently decomposing the computational functions of a specific brain region into the mesoscopic-level anatomy to obtain the HCD that is required for the development of brain-inspired software. In software development, it is common to carry out the design through a process of decomposing the higher-level functions; however, the SCID method also considers consistency with the anatomical structure of the brain.

Furthermore, the decomposition of the functions of the natural brain as if it were an artifact may not yield the desired results. However, as the brain is an organ that has undergone evolutionary selection, its physical mechanisms often serve intended purposes. For example, when computational neuroscience derives "algorithms and expressions" for brain functioning, this action is premised on clear purposefulness.

3.2.1. Process of SCID method

In the SCID method, an HCD that is consistent with the anatomical structure in the ROI is obtained by performing the following three-step process (see Fig. 7).

In step 1, the findings of various studies relating to the cognitive behaviors of humans and animals are used to establish the premise that the SCID method is applicable. In particular, the three processes are performed in parallel. While investigating the anatomical structure around the ROI and registering it as a BIF (1-A), the existence of a component diagram, which we refer to as a provisionary component diagram, that realizes the ROI input/output (1-B) is confirmed. A valid brain ROI and the top-level function (TFL) that it performs are determined (1-C).

In step 2, the TLFs as detailed functional mechanisms are enumerated in any conceivable pattern, with anatomical structures as constraints. Each uniform circuit with a group of neurons of appropriate granularity is first defined to make the structures possessed by the functional mechanism assignable to the

[3] https://ebrains.eu/service/human-brain-atlas/, accessed: 2021-2-26.

Table 2
Advantages of SCID method.

Method	SCID method	Conventional method
Key clues	Structure and TLF (also physiological phenomena)	Neural phenomena correlated with environment (e.g., reward and place cells)
Coverage in brain	Almost entire brain (to the extent that mesoscopic structures are known)	Limited to areas where physiological clues exist
Features	Functional descriptions that are easy to use for development	Phenomenal interpretations that are indirect and software specific

anatomical structure. Specifically, convenient aspects of various cell types are selected, or similar cell type groups are merged. Subsequently, possible candidate functional mechanisms (HCDs) are constructed under the constraint of being included in the connection structures that are contained in the circuit within the ROI of the BIF, whereby each uniform circuit is considered as a component argument.

In step 3, the HCDs that are logically inconsistent according to scientific findings in various fields, such as neuroscience, cognitive psychology, evolution, and biological development, are rejected. Thereafter, the functions of the components and meanings of the connections of the remaining HCDs can be assigned to the BIF.

3.2.2. Advantages of SCID method

In neuroscience, the traditional means of experimentally identifying the function of a neural circuit of interest is as follows. This method identifies neural activity that has an intelligible correlation with an external stimulus, and provides a functional interpretation thereof based on the nature of that stimulus. However, this is only possible if there are brain regions close to the sensor/actuator or neural activity that has a clear correlation with the behavior, such as reward/place cells. In general, it is not easy to obtain interpretable correlations from neural activities that are mixed with various types of external and temporal information in most parts of the neural circuitry of the brain. Thus, the range of neural circuits with functions that can be identified by this method tends to be limited (see Table 2).

The SCID method can be applied to quite a wide area of the brain. This is because the anatomical structure information at the mesoscopic level, which is key to the SCID method, can be obtained from almost the entire brain, including that of rodents (see Section 3.1).

A further advantage of the SCID method is that an HCD is easy to use directly in software development because it is obtained through a process based on the design theory of software development. In contrast, when neural activity phenomena that are correlated with external information are used as a reference for software development, they need to be reinterpreted as a requirement. That is, the functions that are obtained through the traditional phenomenon-based approach (Yamakawa, Arakawa, & Takahashi, 2017) are often indirect information and require preparation for software development.

The first HCD that is developed using the SCID method identifies the site that is responsible for path integration in the entorhinal cortex (Fukawa, Aizawa, Yamakawa, & Yairi, 2020). Subsequently, it is used to identify the meanings of the signals between neocortical regions (Yamakawa, 2020b). Several studies are currently being conducted on the application of the SCID method to study brain regions, such as the brain stem, which is responsible for eye movements (Tawatsuji, Arakawa, & Yamakawa, 2020), the claustrum, and functions including imagination.

Table 3
Metadata for HCD data.

Attribute	Description	Values
n_H: HCD number	Serial number of HCD in this BRA	Natural number
HCD name	Name of HCD	Text
Description	Description of HCD	Text
Implementations	Links to implementations	List of URLs

Table 4
Attributes of data of each HCD.

Attribute	Description	Values
Label (n_H)	Labels for computational functions	Strings
Function (n_H)	Hypothesis of computational function	Text
Projections in use (n_H)	Projections used for computational function	Subset of projections
Comments (n_H)	Comments on computational function	Text

3.2.3. Addition of HCDs to BRA data

The HCD information that is created by the SCID method is registered in the BRA format, as outlined in the following.

As mentioned previously, the BRA consists of BIFs, which are extracted information flows at the mesoscopic level in the target brain region, as well as HCDs, which are hypotheses of the functional mechanisms assigned to the BIFs. Therefore, the functional dependencies that exist in the HCD are described by referring to a part of the projection described in each circuit on the BIF. This is described as the value of the "projections in use (n_H)" attribute in Table 4.

Although only one description of the BIF exists in the BRA data, multiple HCDs may exist. Therefore, these multiple HCDs are managed using Table 3. In this table, for each serial number n_H that specifies each HCD, the "HCD name", "Description", and "Implementations" descriptions can be provided for each corresponding HCD.

The description of the HCD itself is provided by adding the values of the attributes contained in Table 4 for each circuit that is used in the BIF or NBP. These attributes include "projects in use (n_H)" as well as "label (n_H)", "function (n_H)", and "comments (n_H)". Thus, the attribute group corresponding to the description of each HCD is assigned a serial number n_H for the corresponding HCD. For example, in this case, the TLF of the entire ROI in the n'_Hth HCD is described in the function (n'_H) of the circuit that aggregates the entire ROI.

3.3. Adequacy evaluation of BRA

3.3.1. Need for evaluating biological plausibility

When developing brain-inspired software, it is necessary to evaluate the biological plausibility; that is, to estimate the closeness of the implemented brain-inspired software to the reality of the brain as captured by current neuroscience findings.

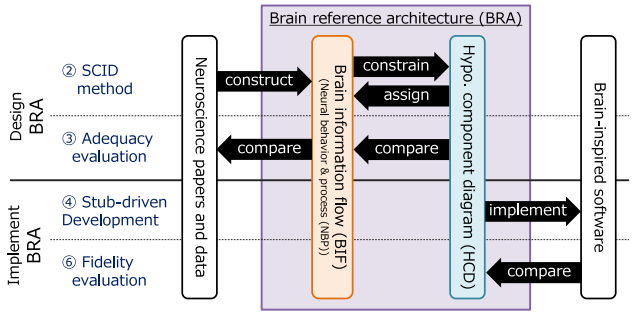

Fig. 8. Creation and evaluation in BRA-driven development. The biological plausibility is evaluated in the direction opposite to that of the creation. In contrast to the SCID method for designing BRAs, the adequacy of BRAs for neuroscientific findings is assessed. Moreover, the fidelity of the software to the BRA is assessed as opposed to the software development. Although this is not depicted in the diagram, in integration development, an HCD is used for both development referencing and fidelity evaluation.

The evaluation of biological plausibility in BRA-driven development involves two methods (see Fig. 8). The first method is the evaluation of adequacy, which estimates the consistency between the existing neuroscientific findings and BRA. The second method is the evaluation of fidelity, whereby the reproducibility of the BRA in brain-inspired software is estimated.

3.3.2. Need for certified registration

The created BRA is used as a functional requirement for reference in software implementation and as a subject for comparison when evaluating the biological plausibility (fidelity). However, the majority of BRA users have little knowledge of neuroscience, and therefore, they cannot determine the trustworthiness of the created BRAs. To ensure the adequacy of the BRA data, a workflow that inspects and certifies the data before they are registered is necessary.

In neuroscience, parallel hypotheses frequently exist that are contradictory but cannot be ruled out. From the perspective of brain-inspired software development, it is not possible to determine which hypothesis is ideal immediately. Therefore, provided that the BRA data meet the inspection criteria, they should be registered even if they contradict other data.

3.3.3. Evaluation of adequacy and inspection criteria

The evaluation of the adequacy is further divided into two parts, as illustrated in Fig. 8.

(1) Adequacy evaluation of BIF

In this process, the consistency of the anatomical structures and neural activity described in the BIF with those described in neuroscientific papers and data is evaluated.

Two main inspection criteria are used to verify that the BIF description is sufficient. The first criterion is that the description element of the structure or phenomenon that is provided in the data submitted for registration is not already registered in the BRA database (novelty). The other criterion is that the element must be directly or indirectly supported by any current neuroscientific findings (authenticity). As a rule, the authenticity of facts is guaranteed by their inclusion in one or more peer-reviewed articles.

(2) Adequacy evaluation of HCD

The functionality of the HCD and its consistency with the BIF are evaluated. The functionality evaluation determines whether the process generated by the behavior of the structured components in the HCD constitutes a mechanism of action that can achieve the goals of the ROI.

The consistency evaluation determines whether the HCD corresponds to the description of the BIF according to three aspects:

1. The dependency structure of the HCD corresponds to the anatomical structure contained in the ROI of the BIF (s-consistency).
2. The behavior of the components within the HCD is consistent with the physiological findings described in the BIF (b-consistency).
3. The mechanism of action for the objective of the ROI is achieved by a chain of actions based on the dependency structure of the components that constitute the HCD (functionality).

3.4. BRA data preparation manual

Details of the BRA data preparation procedure can be found in the "BRA Data Preparation Manual" that is available on the WBAI website.[4] The description in this paper is based on the above manual, which has been made publicly available as of June 2021. As mentioned previously, the BRA consists of the BIF, which is an extracted flow of the information at the mesoscopic level in the target brain region, and the HCD, which is a hypothesis of the functional mechanism assigned to the BIF.

Each BRA datum is handled as a project, which is currently described as the following four sheets in a single Google spreadsheet. The project sheet contains the meta-information for this project. The reference sheet contains a line-by-line list of all bibliographic information used in the project. The HCD sheet (see Table 3) contains the line-by-line meta-information regarding all HCDs specified by the HCD serial number n_H. The main part of the BRA data is the circuit sheet. In this sheet, the BIF, NBP, and HCD information are provided for each row corresponding to a circuit. The list of attributes to be described as BIF and NBP is listed in Table 1. As multiple HCDs exist, the serial number n_H of each HCD is assigned to the attribute group of each HCD, as indicated in Table 4.

4. Development and evaluation using BRA

In this section, we discuss three activities that are associated with BRA use in BRA-driven development. As illustrated in Fig. 8, the development and evaluation that are performed are carried out with reference to the HCD in the BRA; thus, the programmer does not require profound knowledge of neuroscience.

4.1. Stub-driven development

In BRA-driven development, all components are implemented and connected based on the requirements of the HCD associated with a particular task to create brain-inspired software.

In general, machine learning devices often behave differently from the architecture that is imagined at the design phase. The difficulty of controlling this behavior increases rapidly if the system is composed of several machine learning components. The WBA approach uses stub-driven development to address this challenge.

In stub-driven development, a system is constructed during the early stages of development by combining components that do not have a learning function and are described by rule-based processes. Subsequently, the system is improved by gradually replacing each component with machine learning components, so that it approaches the expected behavior in the HCD.

It would be natural to use neural networks for machine learning for implementation in the creation of brain-inspired AI. Depending on the brain organ to be implemented, various neural

4 https://wba-initiative.org/wiki/en/brain_reference_architecture, accessed: 2021-7-2.

networks with hierarchical structures, recursive structures, gating mechanisms, and attention mechanisms can be used. The neocortex, which is compatible with the concept of the Bayesian brain, is also a strong candidate for implementation as a Bayesian network. However, a specific machine learning method is not required by the HCD because it only rules the external behavior of the components.

An integrated execution platform is required as a management mechanism for the computational resources to run and train multiple components in development using a BRA. Candidates for this platform include recent deep neural network platforms, such as TensorFlow, PyTorch, and Keras. Brain-inspired computing architecture is a platform that is developed to consider the asynchronous nature of the brain and other characteristics (Takahashi et al., 2015). Furthermore, an HCD can be constrained and converted into a probabilistic generative model (PGM), SERKET (Nakamura, Nagai, & Taniguchi, 2017; Taniguchi, Nakamura, Suzuki, Kuniyasu, Hayashi, Taniguchi, et al., 2020), and Pixyz (Suzuki, 2021). In recent years, a growing movement known as whole-brain PGM has emerged, which attempts to construct a PGM corresponding to the entire brain (Taniguchi et al., 2021b). The construction of a PGM of hippocampal formation has been initiated (Taniguchi, Fukawa, & Yamakawa, 2021a).

4.2. Fidelity evaluation of software

The biological plausibility of brain-inspired software is evaluated by comparing it with the BIF and HCD in the BRA data. The estimated degree of consistency between the software and BRA is known as the fidelity.

To date, four methods have been explored for the evaluation of fidelity.

- **Structural similarity:** An evaluation of how strongly the static structure of the software matches the BIF in the BRA.
- **Functional similarity:** An evaluation of how strongly the behavior of a particular component that is implemented during the execution of a specific task matches the behavior (e.g., behavior timing) that is designed in the HCD in the BRA.
- **Activity reproducibility:** An evaluation of how effectively the behavior of a certain variable in the internal components of the software implemented according to the BRA reproduces the characteristics of neural activity (such as the activity timing and activity pattern in the corresponding brain region during the execution of a specific task.
- **Performance:** An evaluation of the performance and ability of the software as a whole (integration testing).

Among these evaluation methods, structural similarity and performance are easy to use for the evaluation of the overall software. However, functional similarity and activity reproducibility are useful for unit tests for each component as well as for integration development, as discussed later. Furthermore, it is possible to consider an evaluation method wherein dysfunction states are induced by intentionally destroying/ablating part of the software and comparing it with the brain functioning under conditions such as mental illness or brain injury. Fidelity evaluation (functional similarity, activity reproducibility, and performance) can be performed for behavioral changes owing to learning during task execution by describing how the HCD changes on a specific BIF. However, it is not easy to deal with the cognitive development stage, during which the anatomical structure relating to the BIF changes significantly. Even if the method for describing the BIF could be extended to handle such changes, substantial time would be required for the accumulation of anatomical knowledge to design the BIF in a manner that corresponds to cognitive development.

4.3. Integration development

A particular circuit on the BIF is associated with a component that is included in various HCDs. As noted previously, the HCD is a structure of functions that is decomposed into components to realize TLFs, including tasks. Therefore, even if a component is implemented to realize the same circuit, its function may differ depending on the HCD to which it refers.

However, software needs to be able to apply knowledge to different tasks to reveal its true value as an AGI. To this end, if components exist that correspond to the same brain region in a separate program, integration development is performed by associating and integrating these components. The concept of promoting the integration of components by using brain constraints is known as brain-inspired refactoring. The advantages of BRA-driven development, whereby implementations are performed in response to a common BIF, are also exhibited in such system integration.

4.3.1. Concept of brain-inspired refactoring

A pair of components to be integrated between two implementations can be determined via the BIF in BRA-driven development. Thus, the integration of the entire system can be decomposed into the code integration of each component pair.

As depicted on the left side of Fig. 9, the development of two tasks is performed independently if a BRA is not used as a reference. In task 1, input 1 is assigned to component A, and following processing, output 1 is obtained from component E. Similarly, in task 2, input 2 is assigned to component B, and following processing, output 2 is obtained from component F. The two implementations that are created in this manner are completely different and cannot be merged.

Subsequently, as depicted on the right side of Fig. 9, the case in which two tasks are developed with the BRAs as the constraints is discussed. In this case, components C and D, which are responsible for intermediate processing, are associated with the same circuit on the BIF. Two approaches are typically envisioned to merge components C and D that are contained in different implementations. The first is to compare the fidelity ratings of the two corresponding components and to select the higher-evaluated implementation. The second is to redesign and implement an integrated algorithm that combines the advantages of both.

5. Discussion

5.1. Brain-inspired software development from ontology

5.1.1. Three types of entities in BRA-driven development

The brain can be viewed as both a physical (ϕ) entity and as a device with a function (f) to achieve a goal. Therefore, the ontological perspectives described are relevant for learning from the brain and developing the code (c) that exists as software.

Physical existence (ϕ) has the compositional nature of being constructed bottom-up from a combination of parts. In the brain, no inherent arbitrariness exists in the configurations of the whole from small physical elements, at least at a coarse granularity, and it is expected that it will eventually be uniquely described. Functional (f) existence is obtained through the creation of a subdivision of functions to achieve the purpose that is set as a task. Therefore, a great deal of arbitrariness exists in the manner in which the functional components are combined. The software code (c) also has a compositional nature in the sense that it can be assembled from small literals. However, as the software is developed with the above functional mechanisms as specifications, it reflects the diversity of the functional existence.

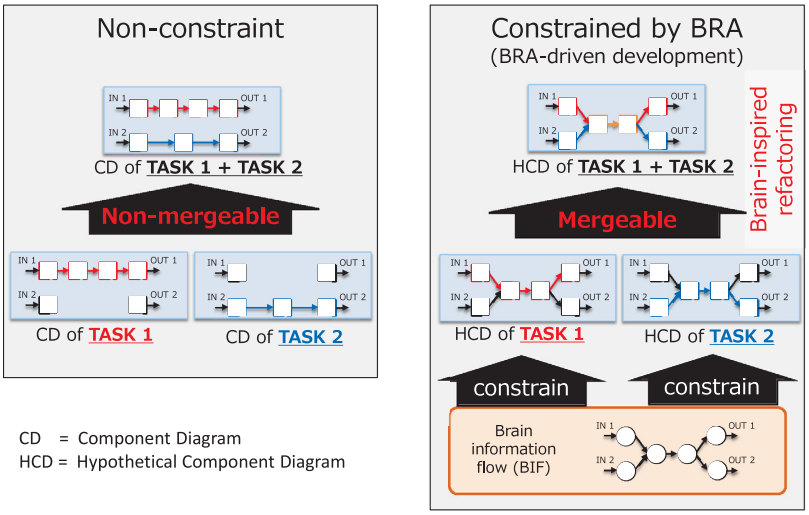

Fig. 9. Components to be merged are specified by BIFs shared by both implementations. In this figure, software development unconstrained by a BIF (left panel) and software development constrained by a BIF (right panel) are compared. Task 1 produces output 1 from input 1, whereas task 2 produces output 2 from input 2. Comparing the component diagrams for these two tasks, it is not easy to identify common points by simply developing them separately, even if the possibility of overlap exists. However, if they are developed with reference to the BRA, the BIF will constrain the HCD that is followed by the software. Subsequently, components C and D, which are responsible for common processing in the two tasks, can be identified as the components to be merged.

Table 5
Entities used in BRA-driven development.

Ontology	Physical (ϕ)	Functional (f)	Coded (c)
Diversity of configurations	Non-arbitrariness	High arbitrariness	
Entities used in BRA-driven development	BRA		
	BIF, NBP	HCD	Software code
Number of entities present	There exists only one entity (1)	Multiple entities can be assigned to a single physical entity (N)	Collection of code implemented based on each functional entity (N)
Scope	ROI	TLF	Software package
Signal	Uniform circuit	Argument of components	
Structure	Anatomy	Networks of components	

Each signal is a minimum description unit. Each uniform circuit is a group of neurons composed of a specific cell type.

The BRA consists of the BIF and NBP, which are physical entities, and the HCD, which is a functional entity (see Table 5). the BIF describes the anatomical structure within an ROI with uniform circuits as the minimal units. The NBPs describe the behavior of the uniform circuits and the physical phenomena that appear as an entire ROI. The HCDs describe the network of components that realize TLFs, with arguments as the minimal units.

In BRA-driven development, entities exist that correspond to each ontology, as follows: First, multiple HCD datasets are associated with one BIF dataset. As mentioned previously, one reason for this is that multiple HCDs are associated with one BIF owing to the arbitrariness of the composition. Another reason is that different functions (tasks) are often associated with the same BIF in HCD design because of the richness of functions realized by the human brain. Software code is typically implemented in various manners based on one of the HCDs.

As the BRA is essentially a reference architecture, it provides a template for brain-inspired software. BRA data reflect the sophisticated mechanisms of the brain; therefore, they are often close to the specification.

5.1.2. Examination of various software development approaches from three ontologies

In this subsection, the entire development process is viewed in three stages: designing the specification, implementing the code using the specification, and evaluating the implemented code. Thereafter, the three stages are compared for four software development approaches in terms of the three ontologies described above (see Fig. 10). Specifically, conventional software, a simulator, conventional brain-inspired software, and BRA-driven development are compared.

First, in the conventional software development approach, the TLF to be realized by the system under consideration is determined. Subsequently, a functional mechanism that can realize the TLF is designed ($f \rightarrow f$) and described as a specification. If the specification is described as a directed graph of components with dependencies, it is known as a component diagram. Thereafter, the software code (c) is implemented based on the specification ($f \rightarrow c$). The evaluation of the developed software includes code review, functional similarity, and performance, to evaluate whether the code conforms to the function ($c \rightarrow f$). The code review verifies whether the content of the code conforms to the functional mechanism. The functional similarity verifies whether each component operates according to specifications (unit test).

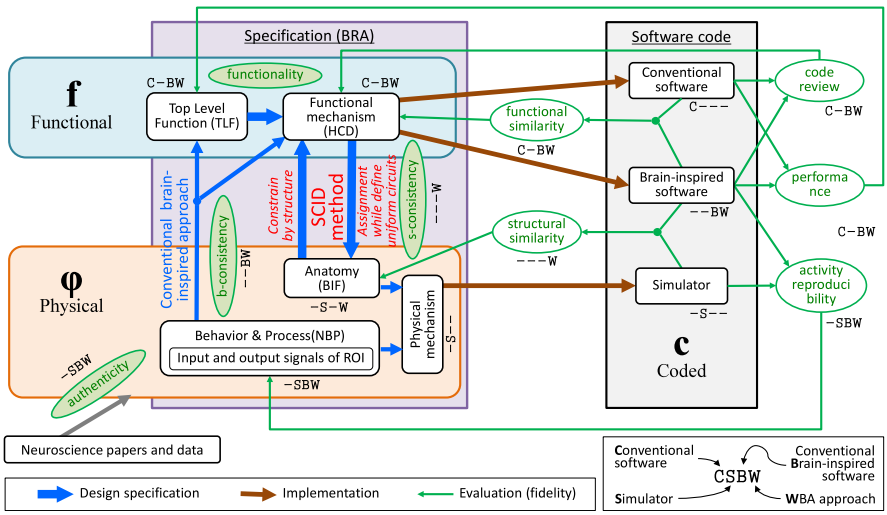

Fig. 10. Four types of development approaches viewed from three different ontologies. In the figure, the character string in "()" is a label indicating the entity that is used in BRA-driven development. The partially hyphenated "CSBW" character string indicates the usage status in each approach: C = conventional software, S = simulator, B = brain-inspired software, and W = WBA approach (BRA-driven development). The character notation part is used in this approach and is not used when replaced with a hyphen character.

In the performance evaluation, the degree to which the TLFs are achieved when the entire code is executed is evaluated.

Second, the simulator development is described. The knowledge (ϕ) of anatomical structures and behaviors/processes (NBP), which are authoritative in neuroscience literature and data, is collected and organized (authenticity). In particular, the input/output signals in ROIs of the brain are important NBP components. The physical mechanism to be realized as software is constructed based on its structure, behavior, and process, and the simulator code (c) is implemented accordingly ($\phi \rightarrow c$). As an evaluation of the simulator that is developed in this manner, we determine the structural similarity and reproducibility of the activities based on whether the code matches the physical characteristics ($c \rightarrow \phi$). Structural similarity verifies whether the structure that is reproduced by the simulator is similar to the anatomical structure of the BIF. Activity reproducibility verifies whether the actions reproduced in the simulator reproduce neural activities and processes in the brain. The evaluation of the input and output signals for ROIs is part of the activity reproducibility evaluation. Formally, this evaluation is the same as the performance evaluation of conventional software (in which the degree of achievement of TLFs is determined).

Third, the approach that is adopted in many brain-inspired software developments is described. As with the simulator described above, NBP findings (ϕ) that are authorized by the literature and data in the neuroscience field are collected and organized. Subsequently, the functional mechanism is designed in such a manner that the TFL can be realized ($f \rightarrow f$) while considering the consistency with NBP (which is referred to as b-consistency). Thereafter, the software code (c) is implemented based on the functional mechanism ($f \rightarrow c$). The evaluation of the neuroscience software developed in this manner is the same as that of conventional software, namely verification that the code matches the function ($c \rightarrow f$). This includes code review, simulation of functions, and performance evaluation. Moreover, the reproducibility of the activity is also evaluated in terms of whether the behavior produced by the code matches the neural activity of the brain ($c \rightarrow \phi$), as in the simulators.

Finally, BRA-driven development is explained. In BRA-driven development, the SCID method is introduced, which makes strong use of brain anatomical knowledge for designing the functional mechanisms in the brain-inspired software described above. As in the previous approaches, knowledge from the literature and data in the neuroscience field is first collected. The anatomical knowledge is organized as the BIF (ϕ), and the knowledge of behavior and processes is arranged as NBP (ϕ). In the design of the functional mechanisms, not only is the functionality to achieve TFL ($f \rightarrow f$) and consistency with NBP (b-consistency) taken into account, but the consistency with anatomical structures by the SCID method (known as s-consistency) is more strongly considered. Functional elements are assigned to anatomical structures by defining groups of neurons with appropriate granularity as a uniform circuit, which is subsequently constructed in such a manner that each uniform circuit is treated as a component argument. The evaluation of the software thus created in BRA-driven development includes all evaluation perspectives of the brain-inspired software described above, such as code review, functional simulation, performance, and activity reproducibility. Furthermore, structural similarity evaluation is performed to determine whether the structure of the code matches the anatomical structure ($c \rightarrow \phi$), as in the simulator.

It is clear that the process for all approaches is as follows: The specification is designed as a mechanism, the code is implemented stringently according to the specification ($f \rightarrow c$ or $\phi \rightarrow c$), and the code is evaluated ($c \rightarrow f$ or $c \rightarrow \phi$). Therefore, to create software that realizes cognitive and behavioral functions similar to those of the human brain, it is very important to construct the specification as a functional and physical mechanism that can realize these functions, while improving consistency with neuroscience findings.

5.2. Related works

At the end of 2020, at least 72 projects relating to AGI development were underway, and in view of the progress from 2017, the total number did not change substantially. However, 15 new projects, most of which were undertaken by private companies, were added during this period (Baum, 2017; Fitzgerald, Boddy, & Baum, 2020).

The following subsection provides an overview of the various AGI development projects from the perspective of the three ontologies depicted in Fig. 10, with a focus on approaches that introduce neuroscience knowledge.

Four large-scale projects were identified in the above survey. OpenAI can be categorized as conventional software development

(Section 5.2.1), the Blue Brain Project and Human Brain Project can be categorized as simulator developments (Section 5.2.2), and DeepMind can be categorized as conventional brain-inspired software development (Section 5.2.3).

5.2.1. AI approach as conventional software development

Naturally, the development of AI does not necessarily involve the use of neuroscience knowledge. In the above categorization, many such AI developments are positioned as conventional software development methods that originate from functional design only.

The projects that flow from symbolic AI are relatively strongly related to design compared to those that depend on learning techniques. Such projects have traditionally taken the form of cognitive architecture research, such as Adaptive Control of Thought – Rational (ACT-R) (Anderson, 2009), SOAR (Laird, Newell, & Rosenbloom, 1987), ICARUS (Choi & Langley, 2018), learning intelligent distribution agent (LIDA) (Franklin, Madl, D'Mello, & Snaider, 2014), non-axiomatic reasoning system (NARS), (Wang, Li, & Hammer, 2018), Sigma (Rosenbloom, Demski, & Ustun, 2016), and Connectionist Learning with Adaptive Rule Induction On-line (CLARION) (Sun, 2016), CogPrime (Goertzel, 2012). Since 2015, an abundance of projects based on deep learning as a technological foundation have emerged, including GoodAI, NNAISENSE, and OpenAI (Brown et al., 2020), which is famous for natural language processing technology.

The main difficulty with this method is that, as mentioned in the Introduction, the design space of the mechanism (or representation or algorithm) inside the computer is vast. That is, as the method of decomposing specific cognitive and behavioral functions is arbitrary, the mechanism by which functional decomposition can realize a particular task is generally not applicable to other tasks. Therefore, it is not easy to design a general-purpose AI (AGI) that can perform various tasks similar to humans.

In certain domains such as natural language processing and image recognition, machine learning using large amounts of data has enabled highly versatile intelligent processing that is comparable to that of humans. However, when the flexible combination of multiple modalities and higher-order thinking such as metacognition are considered, the problem of the design space size remains significant, even with a future increase in computational resources.

5.2.2. Approach to simulate brain behavior

The simulator-type ($\phi \rightarrow c$) approach involves developing code (c) for a simulator that mimics the behavior of the brain from a physical entity, namely the brain (ϕ). This approach is often used in the field of computational neuroscience, which attempts to reproduce phenomena to understand how the brain works.

However, even if the simulation stringently reflects current neuroscience findings, this can only be achieved at the level of reproducing neural activity. That is, the computational function as a whole is yet to be realized (Bostrom & Sandberg, 2008; Markram, 2006). It is necessary to construct a physical mechanism in which the components play appropriate roles and work together organically for the simulator to function as a whole. However, in many cases, current neuroscience knowledge remains limited to the accumulation of fragmentary evidence. Therefore, even if this knowledge were to be reproduced effectively in the simulator, it is inevitable that knowledge gaps will remain that hinder the functional coordination of every component.

It may be possible to fill the above knowledge gaps by repeating various improvements of the simulator. The first improvement is the overall system input and output, similar to normal software development. Thus, the simulator can be improved so that it behaves in the same manner as the results of animal experiments when given the same stimuli. The second method is specific to software that resembles the brain. That is, the behavior of the representations in each component of the simulator during task execution can also be improved to match the neural activity of the corresponding brain region.

One project that uses this approach is Hierarchical Temporal Memory, the aim of which has been to achieve general-purpose computational capabilities in the neocortex since the early 2000s (Hawkins & Blakeslee, 2004; Krestinskaya, Ibrayev, & James, 2018). The "Nengo" project (DeWolf, Jaworski, & Eliasmith, 2020; Eliasmith, 2013), which provides tools to construct the overall cognitive and behavioral functions of the brain at the level of neuronal spikes, also generally adopts this approach, and although it does not explicitly state that it aims to realize AGI, this appears to be the case.

It is expected that the information processing procedures of the brain will be understood in a relatively extensive and complete manner in the future. At this point, the recreation of human-like intelligence through such simulations will be a very promising approach. However, fundamentally, this method does not incorporate the process of designing the mechanism by which the system achieves its goals ($f \rightarrow f$). Therefore, based on the current maturity of the neuroscience field, projects using this approach may lag in achieving human-like cognitive and behavioral functions.

5.2.3. Brain-inspired software development approach

In the brain-inspired software development approach that was explained in the previous section, the process of designing a functional mechanism ($f \rightarrow f$) to achieve a goal is used as the foundation, and neuroscience constraints ($\phi \rightarrow f$) are incorporated. In this manner, this method is expected to fill the gap in neuroscientific knowledge that cannot be compensated for by simulator design, in which mechanisms are designed from physical entities.

Prior to the 2010s, the accumulation of knowledge of comprehensive anatomical structures through connectome studies had not been thoroughly developed. Therefore, traditional brain-inspired software development constrains the design process of functional mechanisms by interpreting the neural activity (NBP) that is observed in neuroscience experiments as a functionality (b-consistency).

In this case, the function is the interpretation of behavior in terms of achieving the goals of the external world. Therefore, in this approach, the neural activities in the sensory and motor areas that are close to sensors and actuators, which are the points of contact with the external world, are easy to interpret and to handle. Occasionally, neural activity may be identified that is clearly related to objects in the external world, even deep in the brain, which can also be handled by this approach.

Therefore, the following developments have taken place by referring to neural activity phenomena in prior neuroscience findings. Neocognitron, which was the starting point for deep learning, was based on the visual cortex. The Vicarious project (George et al., 2018), which aimed to construct an AGI, was also based on the visual cortex. DeepMind, which was founded in 2010 with the aim of building an AGI, consisted of software for hippocampal formation, including grids of cells that could easily interpret spatial dependencies (Banino et al., 2018).

However, neural activity, which can easily be interpreted in terms of achieving external goals, is not widespread throughout the brain. Therefore, the b-consistency-based method of interpreting the function of neural activity phenomena exhibits a major weakness in that it can only be applied to a small portion of the entire brain (c.f. Section 3.2.2).

A further disadvantage of this approach is its low biological refutability, because it is not sufficiently disprovable to reject inappropriate hypotheses by assessing the biological plausibility of the functional mechanism. This weakness stems from the anatomical structures not being mapped to functional mechanisms. The direct problem is the inability to dismiss certain functional mechanisms based on their disagreement with the anatomical structures (s-consistency). Therefore, methods for evaluating functional mechanism hypotheses rely on the consistency between the representation behavior in the code that is implemented based on the representation and the neural activity that is observed in experiments. However, functional mechanisms have not been mapped to anatomical structures in detail. Therefore, the neural activity that is correlated with the representation behavior in the code is allowed to be any neuron in the ROI. Thus, evaluations based on the similarity of activity are ambiguous and do not allow for a rigorous assessment of the validity of certain functional mechanism hypotheses.

However, even if a computational model that is created using this approach does not have sufficient refutability, the possible existence of a mechanism that can realize the input and output of the ROI is proven. This is an important feature that can be used in step (1-B) of the SCID method.

5.2.4. Challenges in using anatomical structures as constraints

As mentioned previously, limitations have existed in constraining the design space of functional mechanisms with the knowledge of neural activity behavior alone. However, in the 2010s, comprehensive studies on anatomical structures, particularly the connectome, started to provide such insights. As anatomical structures are static, they cannot be directly interpreted as functions, but mapping them to a hierarchy of functions can provide strong constraints.

However, two key challenges must be overcome when using anatomical structures to design functional mechanisms. First, the anatomical granularity to be addressed has not been determined. Owing to this indeterminate granularity, it is not possible to write a standardized specification for the design of brain-inspired software, nor is it possible to determine a methodology for its design. This situation arises from the fact that different requirements exist for each position that is involved in brain-inspired software. Implementers may wish to process descriptions at a coarser granularity to reduce developmental costs, but this may inhibit them from taking full advantage of biological constraints. However, if the goal is to make a medical contribution, such as pharmacological effects, a more detailed description is desirable, which may be finer-granular than the most rapid AI can develop. The second challenge is the lack of a methodology that will enable the design of functional mechanisms in the broadest possible range of brain regions, while also using anatomical knowledge at the appropriate granularity.

5.2.5. Uniform circuits and SCID method for using anatomical structures as constraints of function

Therefore, in the BRA-driven development (WBA approach) presented in this paper, (1) the lower limit of anatomical granularity (uniform circuit) to be described is determined, and (2) the SCID method is systematized as a method for designing functional mechanisms at this granularity, as described in the following.

1. **Determination of minimum description granularity (uniform circuit):** The minimum descriptive unit in the brain corresponding to the software argument is defined as a uniform circuit. The candidate brain entities that can correspond to each uniform circuit are assumed to be groups of neurons composed of specific cell types within a particular brain region.
2. **Methodology for designing functional mechanisms based on anatomical knowledge (SCID method):** The proposed SCID method is used for the construction of functional mechanisms ($f \rightarrow f$) to realize TLFs in specific brain regions (ROIs) using anatomical projection structures at a coarser level than uniform circuits (which are relatively well known across a wide range of brain regions) as the main constraint ($\phi \rightarrow f$) (see Fig. 10).

Representations of software code that is implemented based on the functional mechanism, namely the HCD, can be mapped to the anatomical structures of the brain via the BRA. This enables the behavior of a particular representation in the code to be mapped to the neural activity of a specific brain region. Thus, a more detailed assessment of the reproducibility of the activity can be performed.

In step 3 of the SCID method, many of the HCD (functional structure) candidates that are created to achieve the goal are rejected through the determination of the s-consistency mismatch with anatomical structures. Thus, the HCD refinement reflects the refutable nature of the HCD. It is possible for the HCD to be valid at a given time but rejected later owing to new discoveries in neuroscience, which is rather sound for a scientific stance.

In this manner, the data format for the BRA, which is a standard specification for brain-inspired software, can be defined. This open data format will accelerate the accumulation of specifications for the entire brain by allowing multiple teams to share design data. As locally described data begin to aggregate through such sharing, the distributed constraints will interact and the combined constraints will become more powerful. Thus, the design space for brain-inspired software can be rapidly reduced, which will further accelerate the development.

5.3. Roadmap for reaching AGI

As mentioned previously, BRA-driven development consists of a BRA design comprising the BIF and HCD, as well as the development thereof. Given the characteristics of this development methodology, the following five milestones need to be achieved on the roadmap leading to the realization of brain-inspired AGI.

1. **Completion of entire brain BIF:** The first milestone is the construction of a BIF that covers almost the entire brain. It is desirable to build a BIF based on scientific knowledge of the human brain to construct human-like intelligence. However, the knowledge of non-human apes and humans is referenced for the neocortex, whereas the knowledge of rodents is referenced for many other parts. Thus, a chimeric BIF will be constructed. The background of this technical selection is dependent on the degree of neuroscientific knowledge accumulation in major mammalian laboratory animals. Neuroscientific knowledge is the most abundant in rodents and to a lesser extent in humans, and non-human primates will fill this gap. The basic structure of the brain is fairly conserved among mammals, so it is also useful to refer to non-human species. However, humans need to be referenced for the neocortex, which includes areas that carry out human-specific functions such as language.
2. **Completion of entire brain HCD:** The second milestone is the construction of the HCD, which covers typical human cognitive functions. In the construction of the HCD, the target is reward-based decision making and navigation, which are carried out in experiments using animals such as rodents. At this time, the HCD is constructed on the BIF, which is mainly present in rodents. At some stage thereafter, HCDs relating to human-specific functions and tasks such as language tasks, computational/logical tasks,

and metacognition are designed. At this point, the HCD is constructed on the BIF containing the neocortex that is unique to humans. Thus, it is assumed that the HCD of the entire brain will be completed at the stage when the HCD for typical and major tasks that can be handled by humans is described.

3. **Early implementation of WBA:** The third milestone is the implementation of software that integrates almost all parts of the brain according to the BRA. First, computational models, each of which are partial circuits of the brain, are implemented. At this time, the implementation often takes the form of a simple stub, except for the target partial circuit. This enables software that is part of the brain to evaluate the fidelity while performing tasks. When the major parts with a certain quality are almost ready, an integrated WBA system will be constructed. This system will be improved by subsequently assessing the fidelity for various tasks.
4. **Automation of architecture search:** The fourth milestone is the stage in which various brain-inspired architectures that are candidates for AGI can be compared and searched automatically. To achieve this, it is necessary to be able to run and test integrated software for tasks relating to typical human abilities in a virtual environment. The challenge is to promote the automation of biological plausibility (fidelity) assessment using BRA.
5. **Completion of WBA system:** The final milestone is the stage at which software that assembles almost all parts corresponding to the brain neural circuits has been realized so that typical tasks performed by humans can be solved in a manner similar to that of the brain. If the ability to explore computational resource-dependent architectures can overcome the critical point of exceeding the brain-constrained design space size, its completion will occur after a relatively short period. It should be noted that the strengthening of constraints on the design space that accompanies the progress of neuroscience and the improvement of the search ability by increasing the computational resources, which will inevitably continue in the future, will accelerate the arrival of the above critical point.

However, the system may remain insufficient even when all of the computational mechanisms that constitute the brain appear to work together. Regardless, we expect that the lack of technical elements will become apparent once that particular stage is reached.

5.3.1. Evaluation of completion of AGI

It is necessary to decide in advance: "What is the point at which AGI can be determined to have been achieved?" to evaluate the completion of the WBA system in the roadmap. This is also useful as a guideline for the development of the WBA. Therefore, we first consider the assessment of intelligence in the AGI research area.

From a non-anthropocentric perspective, a universal intelligence measure (Legg & Hutter, 2007) exists, which measures the ability of an agent to achieve goals in a wide range of environments. That is, this measure integrates performance in various environments (or tasks) and aims to evaluate the versatility of intelligence independently of human intelligence. However, this measure is theoretical and can only be applied to very small test environments on computers.

In contrast, the WBA approach, which references the brain, aims for human-like intelligence; thus, it would be appropriate to evaluate the completion of AGI from an anthropocentric perspective. The Cattell–Horn–Carroll theory (commonly abbreviated as CHC) has developed a psychological hierarchical classification of abilities relating to general human intelligence. DARPA (2005) reported on cognitive architecture inspired by living creatures, which were broadly categorized as the functional elements of the human brain. Adams et al. (2012) presented a landscape for achieving AGI based on human cognitive development, and discussed the domains of competence and tasks that AGI should encompass. Hernández-Orallo (2010) introduced the universal intelligence test as a set of concepts {space, objects, observation and action, reward} as a class of biologically realistic environments (or tasks). Poldrack, Kittur, Kalar, Miller, Seppa, Gil, et al. (2011) aimed to construct an open knowledge base[5] that integrates and stores the tasks and supporting concepts that are necessary for cognitive neuroscience.

As described above, research relating to AGI evaluation has been conducted for many years using various approaches (Hernández-Orallo, 2017), and its development is expected to continue. If standard AGI evaluation methods are already established near the end of the roadmap, it would be appropriate to use these in the evaluation of the WBA system. However, if an agreed-upon standard list of capabilities is not yet available at that time, it will be necessary to select typical capabilities from among many previous studies, and subsequently, to select a task set for evaluating these capabilities.

In the WBA approach, which references the brain at a relatively detailed level, several constraints can be imposed on the task set. That is, the task set is designed such that the components that are implemented for all brain organs are used in at least one or more of the tasks. In this manner, the completeness of the task set can be improved by checking for leaks from the neural circuitry aspect of the brain.

5.4. Applications of AI systems based on BRA

AI systems that are developed based on BRA can be expected to replicate human cognitive and behavioral capabilities almost exactly. Therefore, BRA offers several practical applications. It enables the construction of an AI that exhibits familiarity with humans when communicating with them. Furthermore, it can be applied computationally to research fields that deal with mental illness and cognitive impairment. Conversely, findings regarding human cognitive impairment may be used for problematic behavior that is observed in brain-inspired AI. Moreover, we believe that this approach can also be used as a computational model that will serve as a device for mind uploading.

6. Conclusions

In this paper, the current WBA approach has been introduced and BRA-driven development to accelerate brain-inspired AGI has been discussed. The BRA includes standardized data that reflect the brain architecture for the purpose of limiting the large design space that is required for a human-level AGI that cannot be grasped by the cognitive ability of an individual. Even developers who do not have a deep understanding of the brain can develop brain-inspired software based on BRAs that are designed by people with expertise in neuroscience. We explained that the BRA is a description consisting of a BIF supported by a mesoscopic neural circuit and an HCD that is consistent with the BIF. Subsequently, to compensate for the lack of neuroscientific findings, we introduced the SCID method, which formulates the creation of an HCD that is consistent with the anatomical structure of the brain. Furthermore, even if a BRA is used for development, individual development results tend to diverge depending on the diversity

[5] https://www.cognitiveatlas.org/, accessed: 2021-7-5.

of the target tasks. To address this problem, integration development is planned, which will move AGI closer to the functioning of the brain. Moreover, we discussed the evaluation of biological plausibility using BRA to prevent the developed software from veering away from the brain.

The main contribution of this study on BRA-driven development, with the following features, is the establishment of a methodology for accumulating data on brain constraints in a form that can be used for software development.

1. Separation of design information: BRA data can be used in various development projects because they are described in a standard format for software development, which is not dependent on a particular development environment.
2. Standardization of description granularity: As a rule, the description of BRA data at a coarser granularity than the mesoscopic level reduces the possibility that the development will focus on details that are unnecessary for the realization of the target cognitive behavioral level.
3. BRA design: The method of designing computational functions according to anatomy (the SCID method) enables BRAs to be created while compensating for the lack of neuroscientific knowledge in a wide range of brain areas.
4. Tolerance of diversity: Even BRAs that contain mutually contradictory HCDs can be registered if they exhibit a certain level of validity, thereby reducing the risk of overly narrowing the considered design space.

The above features of the BRA will provide a foundation for large-scale whole-brain software development as the comprehensiveness of its data increases. Thus, the brain architecture will provide an anchor for the efficient convergence and eventual completion of the development of human-like AGI, whereas the development results in this field tend to diverge at present.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

I would like to thank Naoya Arakawa, Koichi Takahashi, Naoyuki Sakai, Masahiko Osawa, Takashi Omori, Shinichi Asakawa, Kotaro Mizuta, Mei Sasaki, Hirokazu Kiyomaru, Sei Ueno, Hitomi Sano, and Michihiko Ueno for their cooperation in developing the methodology for BRA-driven development. I would like to thank Haruo Mizutani, Hiroshi Okamoto, Yudai Suzuki, Naoyuki Sato, Taku Hayami, and Hiroto Tamura for constructing the basic data for the BIF. I would like to thank Kosuke Miyoshi, Kotone Itaya, Masayoshi Nakamura, Tatsuji Takahashi, Masanori Yamada, Heecheol Kim, Taro Sunagawa, Shion Honda, Yutaka Matsuo, Yuji Ichisugi, Satoshi Kurihara, and Ryutaro Ichise for implementing the related software. I would like to thank Ayako Fukawa, Takahiro Aizawa, Yoshimasa Tawatsuji, Akira Taniguchi, and Ikuko Eguchi Yairi for advancing the SCID method. I would like to thank Haruhiko Bito, Kenji Doya, Tadashi Yamazaki, Michita Imai, Hiroyuki Okada, and Nobuo Kawakami for their discussions on research and development. This work was supported by the Ministry of Education, Culture, Sports, Science and Technology-Japan (KAKENHI Grant Number 17H06315, Grant-in-Aid for Scientific Research on Innovative Areas, Brain information dynamics underlying multi-area interconnectivity and parallel processing), and DWANGO Co., Ltd.

References

Adams, S., Arel, I., Bach, J., Coop, R., Furlan, R., Goertzel, B., et al. (2012). Mapping the landscape of human-level artificial general intelligence. *AI Magazine*, *33*, 25–42.

Amari, S.-I., Beltrame, F., Bjaalie, J. G., Dalkara, T., De Schutter, E., Egan, G. F., et al., OECD Neuroinformatics Working Group, (2002). Neuroinformatics: the integration of shared databases and tools towards integrative neuroscience. *Journal of Integrative Neuroscience*, *1*, 117–128.

Ambler, S. W. (2004). *The object primer: agile model-driven development with UML 2.0*. Cambridge University Press.

Anderson, J. R. (2009). *How can the human mind occur in the physical universe?*. Oxford University Press.

Arakawa, N., & Yamakawa, H. (2016). The whole brain architecture initiative. In *Neural information processing* (pp. 316–323). Springer International Publishing.

Arakawa, N., & Yamakawa, H. 2020. The brain information flow format. In *The 1st asia-pacific computational and cognitive neuroscience (AP-CCN) conference* (p. 0029).

Banino, A., Barry, C., Uria, B., Blundell, C., Lillicrap, T., Mirowski, P., et al. (2018). Vector-based navigation using grid-like representations in artificial agents. *Nature*, *557*, 429–433.

Baum, S. (2017). A survey of artificial general intelligence projects for ethics, risk, and policy. In *Global catastrophic risk institute working paper*. 17.

Bohland, J. W., Wu, C., Barbas, H., Bokil, H., Bota, M., Breiter, H. C., et al. (2009). A proposal for a coordinated effort for the determination of brainwide neuroanatomical connectivity in model organisms at a mesoscopic scale. *PLoS Computational Biology*, *5*, e1000334.

Bostrom, N., & Sandberg, A. (2008). Whole brain emulation: a roadmap. Lanc Univ Accessed January, 21, 2015.

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., et al. (2020). Language models are Few-Shot learners. arXiv:2005.14165.

Choi, D., & Langley, P. (2018). Evolution of the icarus cognitive architecture. *Cognitive Systems Research*, *48*, 25–38.

Clune, J. (2019). AI-GAs: AI-generating algorithms, an alternate paradigm for producing general artificial intelligence. arXiv:1905.10985.

DARPA (2005). *BICA, biologically-inspired cognitive architectures, proposer information pamphlet (PIP) for broad agency announcement 05—18*. Arlington, VA: Defense Advanced Research Projects Agency, Information Processing Technology Office. Obsolete.

de Wit, J., & Ghosh, A. (2016). Specification of synaptic connectivity by cell surface interactions. *Nature Reviews Neuroscience*, *17*, 22–35.

DeWolf, T., Jaworski, P., & Eliasmith, C. (2020). Nengo and low-power AI hardware for robust, embedded neurorobotics. *Frontiers in Neurorobotics*, *14*, Article 568359.

Domingos, P. (2015). *The master algorithm: how the quest for the ultimate learning machine will remake our world. basic books*.

Eliasmith, C. (2013). How to build a brain: A neural architecture for biological cognition. OUP USA.

Erö, C., Gewaltig, M.-O., Keller, D., & Markram, H. (2018). A cell atlas for the mouse brain. *Frontiers in Neuroinformatics*, *12*, 84.

Fitzgerald, M., Boddy, A., & Baum, S. D. (2020). 2020 survey of artificial general intelligence projects for ethics, risk, and policy. *Global catastrophic risk institute technical report*, (p. 20).

Franklin, S., Madl, T., D'Mello, S., & Snaider, J. (2014). LIDA: A systems-level architecture for cognition, emotion, and learning. *IEEE Transactions on Autonomous Mental Development.*, *6*, 19–41.

Friston, K. (2010). The free-energy principle: a unified brain theory? *Nature Reviews Neuroscience*, *11*, 127–138.

Fukawa, A., Aizawa, T., Yamakawa, H., & Yairi, I. E. (2020). Identifying core regions for path integration on medial entorhinal cortex of hippocampal formation. *Brain Sciences*, 10.

George, D., Lavin, A., Swaroo Guntupalli, J., Mely, D., Hay, N., & Lazaro-Gredilla, M. (2018). Cortical microcircuits from a generative vision model. arXiv:1808.01058.

Goertzel, B. (2012). CogPrime: An integrative architecture for embodied artificial general intelligence. *Dynamical Psychology: An International, Interdisciplinary Journal of Complex Mental Processes*.

Goertzel, B. (2014). Artificial general intelligence: Concept, state of the art, and future prospects. *Journal of Artificial General Intelligence*, *5*, 1–48.

Goertzel, B., Lian, R., Arel, I., de Garis, H., & Chen, S. (2010). A world survey of artificial brain projects, part II: Biologically inspired cognitive architectures. *Neurocomputing*, *74*, 30–49.

Hafner, D., Ortega, P. A., Ba, J., Parr, T., Friston, K., & Heess, N. (2020). Action and perception as divergence minimization. arXiv:2009.01791.

Hassabis, D., Kumaran, D., Summerfield, C., & Botvinick, M. (2017). Neuroscience-inspired artificial intelligence. *Neuron*, *95*, 245–258.

Hawkins, J., & Blakeslee, S. (2004). *On intelligence*. Macmillan.

Hernández-Orallo, J. (2010). A (hopefully) non-biased universal environment class for measuring intelligence of biological and artificial systems. In *Artificial general intelligence, 3rd intl conf* (pp. 182–183).

Hernández-Orallo, J. (2017). *The measure of all minds: evaluating natural and artificial intelligence*. Cambridge University Press.
Krestinskaya, O., Ibrayev, T., & James, A. P. (2018). Hierarchical temporal memory features with memristor logic circuits for pattern recognition. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, *37*, 1143–1156.
Kuan, L., Li, Y., Lau, C., Feng, D., Bernard, A., Sunkin, S. M., et al. (2015). Neuroinformatics of the allen mouse brain connectivity atlas. *Methods*, *73*, 4–17.
Laird, J. E., Newell, A., & Rosenbloom, P. S. (1987). SOAR: An architecture for general intelligence. *Artificial Intelligence*, *33*, 1–64.
LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*, 436–444.
Legg, S., & Hutter, M. (2007). Universal intelligence: A definition of machine intelligence. *Minds and Machines*, *17*, 391–444.
Markram, H. (2006). The blue brain project. *Nature Reviews. Neuroscience*, *7*, 153–160.
Meissner, G. W., Nern, A., Singer, R. H., Wong, A. M., Malkesman, O., & Long, X. (2019). Mapping neurotransmitter identity in the whole-mount drosophila brain using multiplex high-throughput fluorescence in situ hybridization. *Genetics*, *211*, 473–482.
Mitra, P. P. (2014). The circuit architecture of whole brains at the mesoscopic scale. *Neuron*, *83*, 1273–1283.
Nakamura, T., Nagai, T., & Taniguchi, T. (2017). Serket: An architecture for connecting stochastic models to realize a large-scale cognitive model. *Frontiers in Neurorobotics*, 12.
Negishi, S., Hayami, T., Tamura, H., Mizutani, H., & Yamakawa, H. (2019). Neocortical functional hierarchy estimated from connectomic morphology in the mouse brain. In *Biologically inspired cognitive architectures 2018* (pp. 234–238). Springer International Publishing.
Oh, S. W., Harris, J. A., Ng, L., Winslow, B., Cain, N., Mihalas, S., et al. (2014). A mesoscale connectome of the mouse brain. *Nature*, *508*, 207–214.
Petersen, S. E., & Sporns, O. (2015). Brain networks and cognitive architectures. *Neuron*, *88*, 207–219.
Poldrack, R. A., & Gorgolewski, K. J. (2017). OpenfMRI: Open sharing of task fMRI data. *Neuroimage*, *144*, 259–261.
Poldrack, R. A., Kittur, A., Kalar, D., Miller, E., Seppa, C., Gil, Y., et al. (2011). The cognitive atlas: toward a knowledge foundation for cognitive neuroscience. *Frontiers in Neuroinformatics*, *5*, 17.
Pradeep, A., Knight, R. T., & Gurumoorthy, R. (2013). Neuro-informatics repository system.
Rosenbloom, P. S., Demski, A., & Ustun, V. (2016). The sigma cognitive architecture and system: Towards functionally elegant grand unification. *Journal of Artificial General Intelligence*, *7*, 1–103.
Sasaki, M., Yamakawa, H., & Arakawa, N. (2020). Construction of a whole brain reference architecture (WBRA). In *International symposium on artificial intelligence and brain science* (p. 31).
Sun, R. (2016). *Anatomy of the mind: exploring psychological mechanisms and processes with the clarion cognitive architecture*. Oxford University Press.
Suzuki, M. (2021). Pixyz. https://pixyz.io/. Accessed: 2021-7-28.
Takahashi, K., Itaya, K., Nakamura, M., Koizumi, M., Arakawa, N., Tomita, M., et al. (2015). A generic software platform for brain-inspired cognitive computing. *Procedia Computer Science*, *71*, 31–37.
Takahashi, Y., Schoenbaum, G., & Niv, Y. (2008). Silencing the critics: understanding the effects of cocaine sensitization on dorsolateral and ventral striatum in the context of an actor/critic model. *Frontiers in Neuroscience*, *2*, 86–99.
Taniguchi, A., Fukawa, A., & Yamakawa, H. (2021a). Hippocampal formation-inspired probabilistic generative model. arXiv.
Taniguchi, T., Nakamura, T., Suzuki, M., Kuniyasu, R., Hayashi, K., Taniguchi, A., et al. (2020). Neuro-SERKET: Development of integrative cognitive system through the composition of deep probabilistic generative models. *New Generation Computing*, *38*, 23–48.
Taniguchi, T., Yamakawa, H., Nagai, T., Doya, K., Sakagami, M., Suzuki, M., et al. (2021b). Whole-brain probabilistic generative model towards cognitive architecture for developmental robots. arXiv.
Tawatsuji, Y., Arakawa, N., & Yamakawa, H. (2020). Knowledge representation for neural circuits subserving saccadic eye movement based on a brain information flow description. In *International symposium on artificial intelligence and brain science* (p. 45).
Triplett, J. W., Owens, M. T., Yamada, J., Lemke, G., Cang, J., Stryker, M. P., et al. (2009). Retinal input instructs alignment of visual topographic maps. *Cell*, *139*, 175–185.
Wang, P., Li, X., & Hammer, P. (2018). Self in NARS, an AGI system. *vol. 5*, In *Frontiers in robotics and AI* (p. 20).
Williams, M. E., de Wit, J., & Ghosh, A. (2010). Molecular mechanisms of synaptic specificity in developing neural circuits. *Neuron*, *68*, 9–18.
Yamakawa, H. (2020a). Attentional reinforcement learning in the brain. *New Generation Computing*.
Yamakawa, H. (2020b). Revealing the computational meaning of neocortical interarea signals. *Frontiers in Computational Neuroscience*, *14*, 74.
Yamakawa, H. (2020c). Towards a qualitative evaluation of biological plausibility for brain-inspired software. In *The 1st asia-pacific computational and cognitive neuroscience (AP-CCN) conference* (p. 0031).
Yamakawa, H., Arakawa, N., & Takahashi, K. (2017). Reinterpreting the cortical circuit. In *Architectures for generality & autonomy workshop at IJCAI. Vol. 17*.
Yamakawa, H., Arakawa, N., & Takahashi, K. (2020). Whole brain reference architecture to evaluate biological plausibility of human-like artificial intelligence. In *International symposium on artificial intelligence and brain science* (p. 30).
Yamakawa, H., Osawa, M., & Matsuo, Y. (2016). Whole brain architecture approach is a feasible way toward an artificial general intelligence. In *Neural information processing* (pp. 275–281). Springer International Publishing.