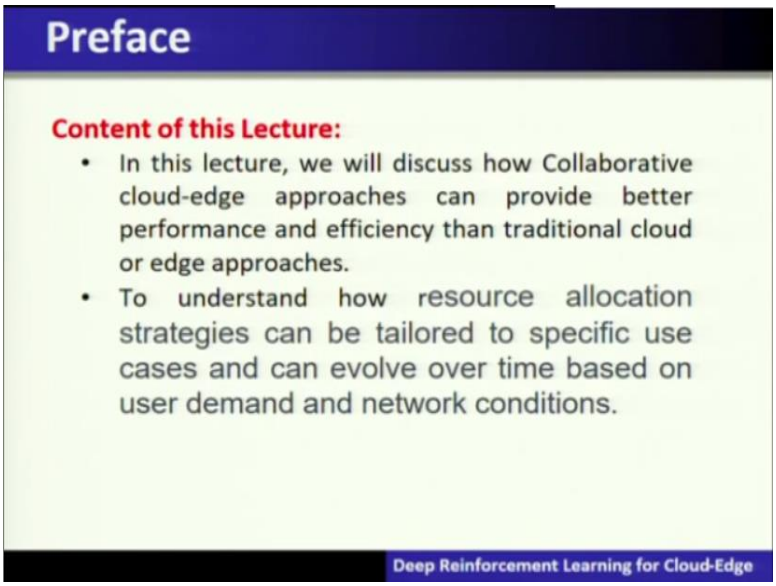
Foundation of Cloud IoT Edge ML
Professor Rajiv Misra
Department of Computer Science and Engineering
Indian Institute of Technology Patna
Lecture 09
Deep Reinforcement Learning for Cloud-Edge

Hello, welcome all to this particular lecture. I am Doctor. Rajiv Misra, working at IIT, Patna. The topic of this lecture is Deep Reinforcement Learning for Cloud-edge system.

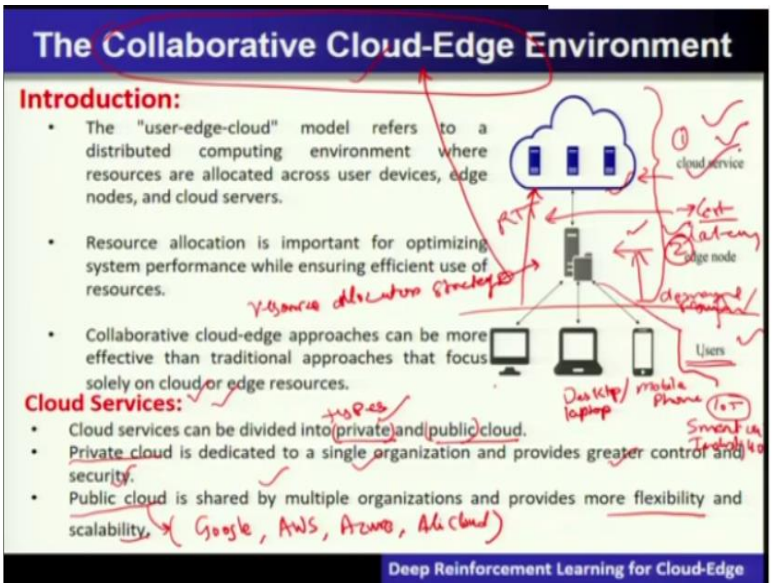
(Refer Slide Time: 0:26)



Content of this lecture. In this lecture, we are going to cover how the collaborative environment cloud-edge can provide the better performance and efficiencies compared to the traditional cloud or the edge approaches. So, here we will explain you about the model that is a cloud-edge model, which brings into the edge computing, how the resource allocation is to be done.

And, we will show that in this particular scenario, to have this optimal cost. And this cost means that various parameters of optimization, which is possible in the edge cloud system, that is latency, then bandwidth cost, and the performance. These are, and also the user devices, batteries. We will also understand how the resource allocation using all these as a part of the strategy can tailor to specific use cases and you all over to the time based on the user's demand and the network conditions. These are few things which will be covered in this part of the lecture.

(Refer Slide Time: 1:40)



So, let us understand this cloud-edge environment that is, a collaborative cloud-edge environment, which is considered as the model for this kind of problem formulation. So, let us introduce about this collaborative cloud-edge environment. So, the user-edge-cloud model, which is referred to a distributed computing environment, where the resources across the user devices like user nodes or mobile phones, edge nodes and cloud servers requires the resources to be allocated.

So, therefore, once we consider that resource allocation, so we can also simplify the resources in the form of virtual machines. That is, the resources are, we mean that resources are the computing requirements. Then memory usage or the network bandwidth. So, this together we can say that resources are nothing but virtual machine environment. That is, all the resources are virtualized.

So, the resources allocation becomes an important for optimizing the system performance and also ensuring the efficient use of resources. So, the collaborative edge approaches can be more effective than traditional approaches that focus solely on cloud and edge resources. Let us understand this collaborative cloud-edge environment through this particular figure. Now, here comes the users. So, users are of different nature.

For example, let us say it is a mobile phone. As a device, we can say that this is a user or it is a desktop or it is a laptop, or it can be any IoT device as well. So, IoT devices, as you understand, they are equipped with the sensors and often deployed in the environment for this monitoring. For example, a smart city environment or industry 4.0 environment. So, these type of environment you will see the devices are, they are nothing but an IoT devices.

So, let us understand that this particular segment, which is nothing but we consider it as the users. So, these users will generate the demand for the resources. This demand is for the resources, and this resources will be met with the help of collaborative cloud-edge environment. So, let us understand this collaborative cloud-edge environment.

So, here the edge node, you know that comprises of the data centre or the computing devices, which is very close to these users nodes. They are called edge nodes. We have already defined this edge node and also we will define later on. Then above the edge node, there exist a cloud environment. So, there is a continuous interaction, whether the resources will come from the edge node or the resources will come from the cloud node. It has that cost implication.

If you get the resources from the edge node, then the cost will be less. When you say cost, it means various factors. The, let us say that latency is a cost function. So, latency means these particular requests will come very close to the devices and therefore the cost that is the latency will be very low if the, if that computing resources comes from the edge node.

Now, you know that edge node also has a limitation of the resources availability. Therefore, not all demand can be met by the edge. So, therefore, this condition, when edge has exhausted all the resources and still more resources are needed by this industrial use case, then these resources has to be met directly from the cloud.

So, in that case, this particular cost function will have that implication that it will have more latency and therefore, this latency is due the round rep time sending the requests to the cloud, performing the computing at the cloud and getting the result back contains the round rep time. So, therefore it incurs more cost, but it will provide the, the resources, whatever is needed beyond the edge nodes or beyond the user devices. So, that is a possibility in this environment.

So, this collaborative cloud-edge environment provides, abundance of resources while optimizing various cost functions. That is where we are going to cover over here. So, in a dynamic environment, when or how much of these particular resources are to be met by the edge and how much resources are to be made by the cloud becomes an intelligent decision making or a strategy. So, that is what is the title of this particular lecture; resource allocation strategy.

So, resource allocation strategy will guide this environment that is the cloud-edge environment for how the resources are going to be allocated with the minimum cost implications. So, let us understand different components. The cloud service is the first component. Then we are going to cover the edge node, and then we will also talk about this environment.

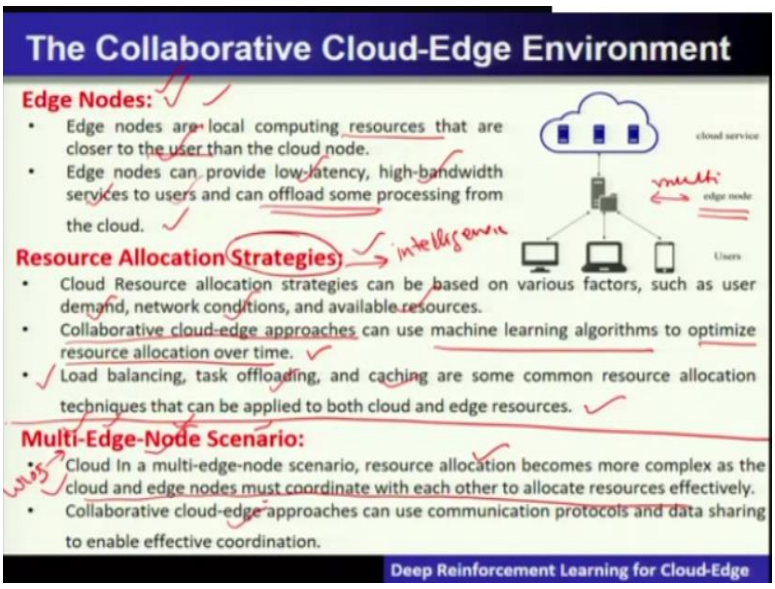
So, let us start with the cloud services. So, cloud services as a part of collaborative cloud-edge environment can be divided into two parts or a two types. So, the first type is called a private cloud. The other type is called a public cloud. So, when you talk about a private cloud, so private cloud is nothing but a dedicated single organization dedicated to an organization and provides a greater control and security.

Therefore, it has these kind of advantages when you do the computing at the private cloud. So, private cloud, as you know, that it is a dedicated to a single organization. So, it is not providing the elastic amount of resources on the premise.

So, another type of cloud is called a public cloud. Public cloud is shared by multiple organization and is available for the public. When you say public means, anyone can access, who is ready to support this, pay as you go by model.

So, there are various, it can be shared by multiple organization and also the public cloud provides more flexibility and scalability. So, example of a public cloud is, the Google cloud and Amazon AWS cloud or Azure or Ali Cloud. These are some of the examples of the public cloud. So, we have explained one component called cloud service.

(Refer Slide Time: 9:39)



Now, another component let us understand about the edge node. So, this is the second component, we are explaining about the edge nodes. So, edge nodes are local computing resources that are very close to the user compared to the cloud. So, this is very close by, close to the, so computing very close to the user is often called as the edge nodes. So, as nodes, if, let us say that computing resources come from the edge node. So, these edge nodes can provide low latency, that means the roundtrip is very less.

Why? Because it is very close to the user device and also it provides a high-band services, high bandwidth services. Since it is very close, therefore, the access, bandwidth is also going to give a much better high quality to the users and can offload some of the processing from the cloud to the edge node. So, instead of doing these kind of computing at the cloud, it can be done at the edge. Therefore, so it can offload these computing or processing from the cloud to the edge. So, these are the edge, about the edge nodes.

Now coming about the resource allocation strategy, whether to be resources, whether it is to coming from edge node or from the cloud node. So, this requires a resource allocation strategy. So, the cloud resource allocation strategy can be based on various factors such as user demand. So, the demands of that resource from the users, network conditions and availability of these resources. There are three different important factors which are going to affect this cloud resource allocation as far as the strategy is concerned.

So, if you talk about the collaborative edge approaches, therefore this strategy is going to be very intelligent. If a human is involved to decide this particular strategy based on this (u) user demand, network condition and available resources and also optimization conditions. Therefore, humans are good at it. So, if you replace the humans with an agent to take this particular decision. So, this particular collaborative edge, cloud environment, that is the resource allocational strategy, has to be met with some amount of intelligence like humans.

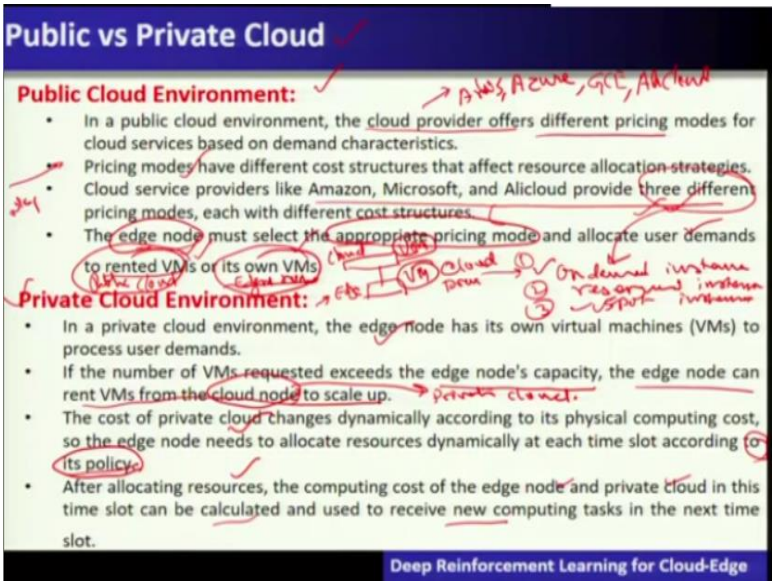
Therefore, we are going to use, we will show here in this lecture that we can use or we can apply machine learning algorithms and therefore, it can optimize this resource allocation over the different time period. Now, besides this, there is also in this way that these resource allocation strategy also has to support, the network load balancing, task offloading and caching. So, therefore, these kind of objectives for load balancing, (top), task offloading and caching are some of the very common resource allocation strategy that can be applied to the both cloud as well as the edge and resources.

Now let us consider a more general or more general conditions. So, that is called multi-edge and multi-edge node conditions. So, here in this example, what we have assumed is a single node, but nevertheless, it can be extended to a multi-edge nodes. So, when you are going to to go for the multi edge node, that means you are going to generalize the first condition that is instead of a single node, it is having a multi-edge node. We mean that the resource allocation will become more complex.

So, which of these multiple edge nodes is going to provide you the resources. So, that also will be a part of decision making. So, that means instead of single node, multiple nodes exist, and now there is a multiple possibilities of getting the resources from the different edge environment. So, therefore, this resource allocation becomes quite complex as the cloud and edge node must coordinate with each other to allocate the resources effectively.

So, we will show in this particular use case with a single edge node based resource allocation strategy that is called cloud-edge environment. Nevertheless, without loss of generality, you can extend this for this. The same approach which we are going to cover can extend for multiple multi-edge node scenario as well. So, the collaborative edge, approaches cloud-edge approach can use different communication protocols and data sharing to enable effective conditions.

(Refer Slide Time: 14:40)



So, let us understand about the other aspect that is after the cloud, there is aspect of public and the private cloud. So, public cloud environment, which is a part of a cloud environment in a cloud-edge environment. So, regarding that, let us see about the public cloud environment. So, public cloud environment, which comes as a offering from different cloud

providers. Examples we have already told, like Amazon AWS, and Microsoft Cloud is called Azure and Google Cloud Engine and Ali Cloud.

So, there are different type of cloud providers or cloud services which are available, and they often offer different pricing modes for providing these cloud services with an on-demand characteristics. So, pricing modes have different cost structures that affect these resource allocation strategies. So, we are going to see how we are going to incorporate multiple pricing modes. So, as, a simplified model we are going to use, but nevertheless, these cloud, provider provides different type of cost or apprising models, which can be extended as the general case without loss of, generality here in this particular use case.

So, this, we have said that we have taken a common type of pricing model, which is prevalent across most of these cloud, vendors, which are of three different price modes. So, three different price modes, which we are now going to discuss, or which we are going to cover in this model are on-demand instances, then reserved instances, and then spot instances.

So, on-demand instances or spot instances are that kind of cloud services or the cloud resources, cloud pricing, which says that the resources are not resolved upfront, rather whatever is, coming, as the user demand and that particular instances will be resolved based on that called on-demand and spot instances. And they have different, pricing modes.

Third pricing mode is called reserved instances. So, reserve instances, whether you use it or not, these cloud instances are in the form of virtual machines, are in the form of cloud resources are already put in place or already reserved. So, it has a different pricing model. So, here we are going to discuss or we are going to consider three different pricing modes, which are common across all most of these cloud providers.

Now, as far as the edge node is concerned, as node has to select the appropriate pricing mode and allocate the user demand to the rented virtual machine or its own virtual machines. So, you mean to say that here you have an edge node, then you have the cloud node, then you have a cloud. So, the edge node has to now select whether the virtual machines will come from the edge or virtual machine has to come from the cloud. That means the resources, either it has to come from the edge or it has to come from the cloud, and it has to decide.

So, as you note, has to decide about when it is going for the, virtual machines to be allocated from the cloud, what is the appropriate pricing mode, and then allocate it, based on this user demand. So, that will be called rented virtual machine. If the virtual machine, if the resources

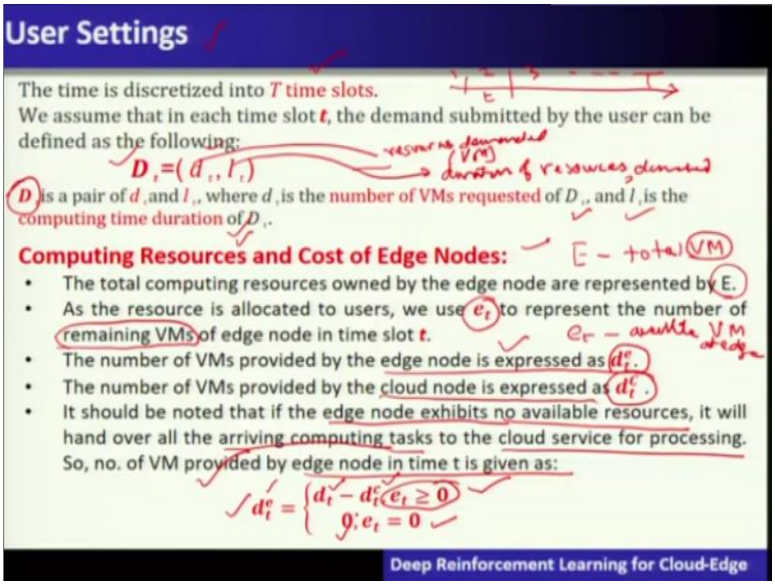
come from the public cloud, and it is called own virtual machine, if the resources come from the edge node. So, you know that if it is coming from, very close by, whether it is a private cloud or edge node, then it is called own virtual machine is called edge node. And if it is rented, that means it is coming from the cloud environment.

So, let us understand about the private cloud environment. So, in a private cloud environment, the edge node had its own virtual machines to process user demand. If the number of virtual machines requested exceeds the edge nodes capacity, the edge node can rent the virtual machines from the cloud node to scale up. Now, this cloud node will be the private cloud that is only accessible to the organization need not for other organizations or other public. That is why it is called private cloud.

So, the cost of private cloud changes dynamically according to the physical computing cost. So, the edge node need to allocate resources dynamically at each time slot and according to its policy. So, this is, we are going to see, how to formulate this particular policy so that it eventually, that particular policy of a resource allocation eventually will optimize, this entire cloud-edge environment or the cost function.

So, after the allocation, according to the strategy or a policy, the computing cost of the edge node and the private node in this time slot can be calculated and used to receive the new computing task in the next time slot.

(Refer Slide Time: 20:41)



So, let us understand, the problem formulation under this cloud-edge environment, starting with the user setting. So, the time is flowing or is time is discretized into the time slots. So,

time is slotted as 1, 2, 3, 4, and so on up to t time slots. So, we assume that in each time slot t the demand which is submitted by the user can be defined using this particular equation, we call it as a demand. So, user demand. So, demand DT is a pair of, small dt and It. So, DT is the number of virtual machines which are requested by, by the user demand at a time stands t

And, not only that virtual machine which is requested, which is, defined as a small dt of, DT also has another pair with a small dt associated that is called It. So, It is nothing but the duration for which this virtual machine is required. So, they are not only the virtual machine that is the resources, not only these resources, but also the time. So, this is nothing but the resources demanded in the form of virtual machines, but also for what duration. Duration of the resources demanded.

So, therefore, this DT that is the demand is a pair of number of virtual machines requested at a time slot at a particular time slot t, and also the duration for which that particular resources will be required. So, this is, going to define that is the demand. So, let us understand that once the demand is properly defined, and then we will formulate about the computing resources and the cost of these edge nodes.

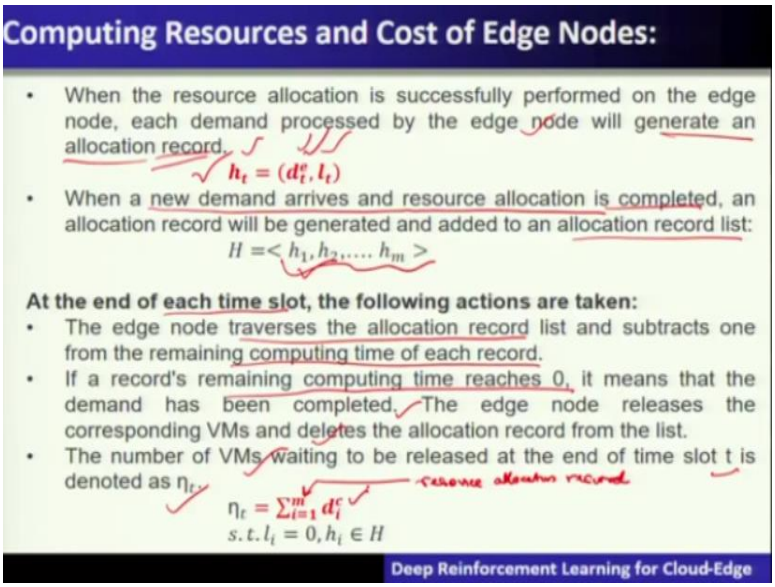
So, let us see that the total resources, which are owned by the edge node is represented by capital E. Now as the resources is allocated to the user. So, we use a small e to represent the number of virtual machine which are remaining of the edge node at the time t. So, when you say remaining virtual machines, that means they are available virtual machines, which is represented by et. So, the total virtual machines, total VM at the edge node represented by capital E and small et is represented by the available virtual machines at the edge.

So, the number of virtual machines, which are provided by the edge is expressed by this, d of e at a particular time t. So, the number of virtual machines which are provided by the cloud node is also represented in the similar manner d c, at time t. Now, it should be noted that if the exhibits no available resources, that means if the edge node does not have the available virtual machines, then in that case, we will assume that it will hand over or it will ask that means, it will lease all the computing, demand from the cloud service for doing this cloud-edge computing or resource allocation.

So, the number of virtual machines which are provided by the edge node in a time t is written by this equation that is d of e that is, the number of virtual machine which are provided by the edge node is nothing, but that is the demand minus which is being met by the cloud. So, the remaining will be met by the dt, if the number of virtual machines are, is the available virtual

machines is more than 0, it is non-zero, then this equation will hold. And if the virtual machines are not available, then this demand, from, that is a virtual machines which are to be provided by the edge node, will become 0 in that case.

(Refer Slide Time: 25:07)



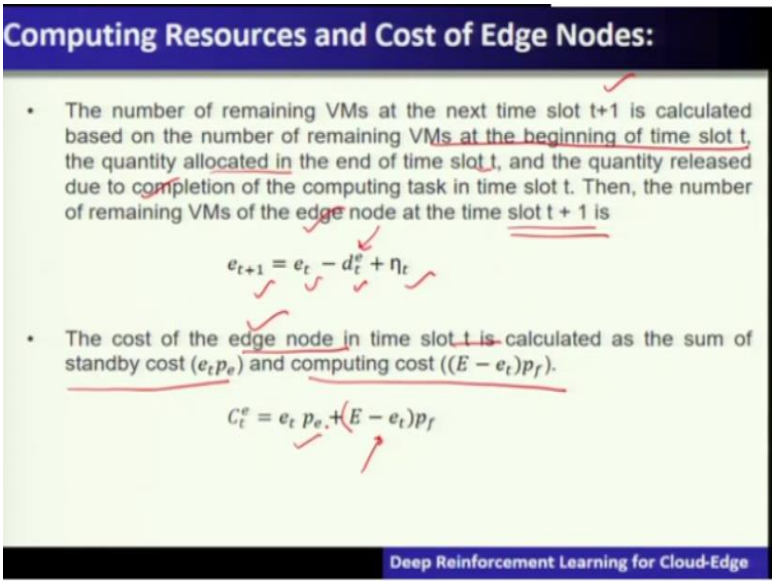
So, having understood these equations, so let us see that when the resources, when the resource allocation is successfully performed on the edge node, so each demand will be processed by the edge node will generate the allocation, record. So, let us understand, formalize this allocation. So, this allocation is, to be in the form of allocation record. That means the, the allocation, resource allocation, if it is decided by some strategy, then it will be represented by this resource allocation record.

So, resource allocation record is h that is nothing but d e means that what which means, which is being made available from the edge node, and It means the time for which these virtual machines will be provided by the edge nodes. So, when a new demand arrives, let us say, and the results allocation is completed, and allocation record will be generated and added to the allocation record list. So, this particular example shows that capital H requires different allocation is, is to be collected. So, resource allocation is nothing but this resource allocation record.

Now, at the end of each time slot, the following actions are taken so as node traverses, this edge allocation record and subtract one from the remaining computing time of each record. Now, if the record's remaining computing time reaches 0, it means that the demand has been completed, the note releases the corresponding virtual machines and deletes the allocation record from the list.

So, this particular list keeps on updated, because the computing in the cloud-edge environment is a dynamic in nature. So, the number of virtual machines which are waiting to be released at the end of time t is noted by eta. So, eta, is nothing but the submission of all m as we have seen this, particular resource allocation record m is resource allocation record, it is the submission of all the virtual machines, all the resources which are allocated out of a, from the cloud.

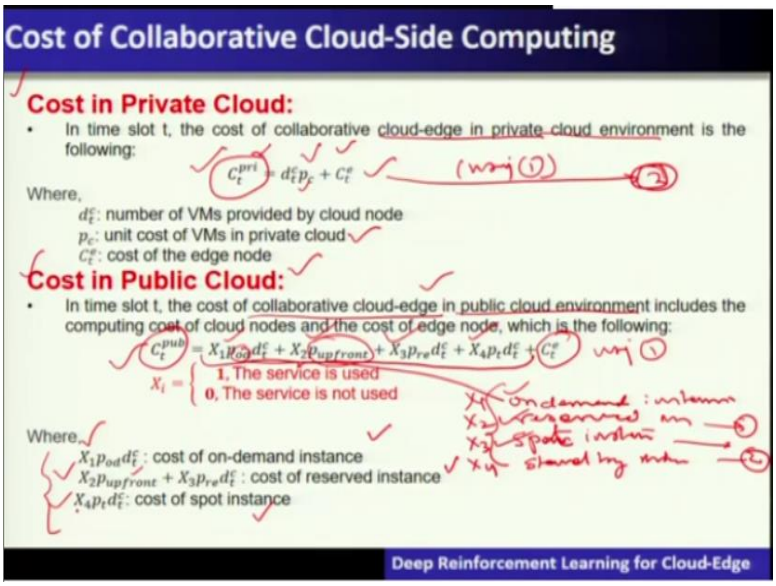
(Refer Slide Time: 27:22)

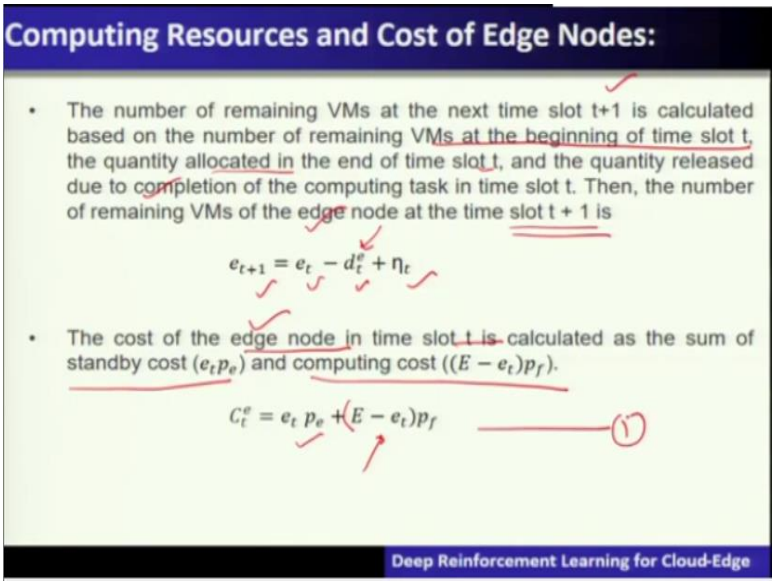


Now the number of virtual machines at the next time slot t1 is calculated based on the remaining virtual machines at the beginning of a time slot and the quality allocated at the end of, the time slot, T. So, the quantity released due to the completion of computing task in time t. So, then the number of remaining virtual machines of the edge node at the time, times slot t, plus 1 or the next time slot is given by e t plus 1 is equal to e t minus d of e and that means at the time t, the virtual machines from the edge, which has completed and, new t.

So, the number of remaining virtual machines at the, at the edge node at time, t plus 1 is given by this. So, therefore, the cost of edge node at the time t, you can calculate as the sum of standby cost and the computing cost. So, the computing cost is shown over here, so this is the total number of virtual machines, which are available by the edge nodes. And this is number virtual machines which are provided. So, this will become that particular, the cost of the edge node and, this is time spend by cost. So, the virtual machines which are already, allocated and resolved.

(Refer Slide Time: 28:46)





So, let us understand that in this edge environment, if you add a cloud, so there are two possibilities. So, one is the private cloud. So, if let us say that the demand is not met wholly by the edge node, often, partially by the edge node and partially by the private cloud. So, therefore, the cost structure will change. So, at time t the collaborative edge cloud in using the private cloud environment, the cost is going to be following.

So, the cost of using the private cloud C of a private cloud is nothing but the cost of the cloud. And, that is a demand met by the cloud and, per unit cost. So, that is what is written. And then the cost of the edge which we have calculated in the previous slide. So, let us refer that, let us say that this is the equation 1, and using this particular equation 1, you are going to get this equation 2, that is the cost of.

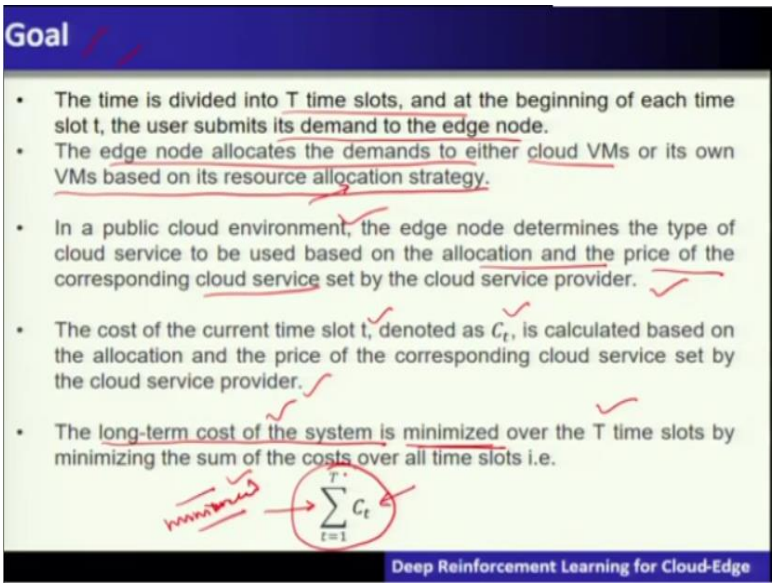
Now let us understand the cost if you use, not the private cloud, but the public cloud. So, the edge and the public cloud, if let us say so. So, edge plus public cloud in a collaborative, cloud-edge environment with the public cloud environment, is included, then the cost, will be the following. So, the cost of using the public cloud that is C of, public cloud is nothing but, it will be the cost of using the, resources from the edge using equation number 1 that we have already explained. And then if the, if the demand is not met wholly by the edge, then it will be coming from the public cloud.

Now, public cloud has different type of services to be offered. So, for example, if it is on demand service, on demand instances, then it has a cost, that means how much virtual machines of this type that is on demand instance type is used. That is P of od. And X1means that this service is used, otherwise 0 will be fed. Now, the second type of service is called, is spot instance. Third type of service is called reserve instance.

So, all are mentioned were here, X1 is on-demand, X2 is reserved instance, X3 is spot instance, and X4 is some other, some other type of instance. So, all these instance is comprises of, is stand by instance. So, all these instances are represented by X1, X2, X3, and X4. So, this will become the cost function if, let us say that a collaborative edge cloud using the public cloud environment is being considered. So, that is what is explained over here.

So, X1 means that the cost of on-demand instance, X2 means that the cost of reserved instance, there are two types of reserve instance that they have considered. And third one is the cost of spot instance. So, there are four different types which are mentioned over here, I have already explained it.

(Refer Slide Time: 32:36)

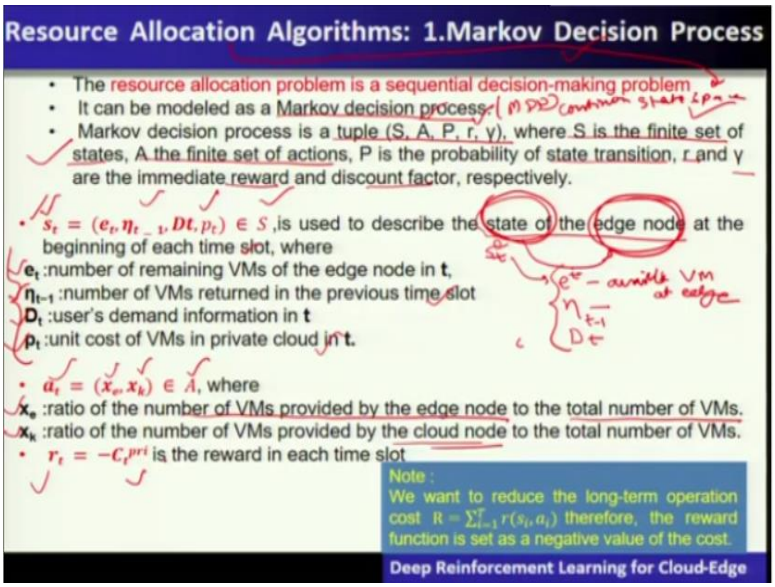


So, having understood this cloud-edge collaborative environment. So, let us see what is the goal of resource allocation strategy? So, we have already considered the type is slotted into the, into the t time slots. And at the beginning of his time slot, the user submits his demand to the edge node. Now, the edge node has to allocate, the resources as per the demand, from either the cloud virtual machines or its own virtual machines. And this has to be decided using the intelligence that is called resource allocation strategy.

So, in a public cloud environment, the edge node determines the type of cloud service to be used based on the allocation and the price of the corresponding cloud service, which is set by the cloud service provider that we have already considered in the cost function. So, the cost of the current time slot t, which is denoted by Ct, is calculated based on the allocation and the price of the corresponding cloud service set by the provider. So, the long-term cost of the system, the goal is to minimize over the time.

So, the cost is to be minimized or the t time slots by minimizing the summation of the costs over all time slots. And this is written over here. So, this has to be minimized. That is the sum of the cost over all time slots has to be minimized. So, this is the goal of resource allocation.

(Refer Slide Time: 34:22)



Resource allocation algorithms, formulation using Markov decision process. So, the resource allocation process or a problem is a sequential decision making problem. Now, when it is a sequential decision making problem, that means the state space is continuous. So, so for a continuous state space. So, for a continuous state space, which often is to be considered as the resource allocation, it can be modelled using Marco decision process called MDP

To model this problem using Marco decision process, we can describe this MDP formulation with a tuple called S, A, P, r and, gamma. So, S is a finite set of states, and A is a finite set of actions, P is the property of the state transition, r and, gamma is the immediate reward and the discount factor. Whereas, this particular formulation, is explained, regarding when talk about the states finite set of states in capital S, what is that state means?

So, S from that, from that tuple, that is finite set of states. States is described as the state of a edge node at the beginning of each time slot. So, the state we talk about the state of the edge node that we have to pay attention. So, states, when we say state st, so, so it is a state of an edge node we are talking about. And let us see that how we are going to model. So, states of an edge node, we have already explained, it is nothing but the number of remaining virtual machines. So, e of t is the available virtual machines, available virtual machines at the edge.

And then eta t, is, eta is the number of virtual machines which are returned. So, which are returned from the previous times, slots t minus 1 is the number of virtual machines which are returned from the previous times slot. Dt is the user's demand at time t and Pt is the unit cost of virtual machines, if it is being leased from the private cloud in time t. So, this, all these

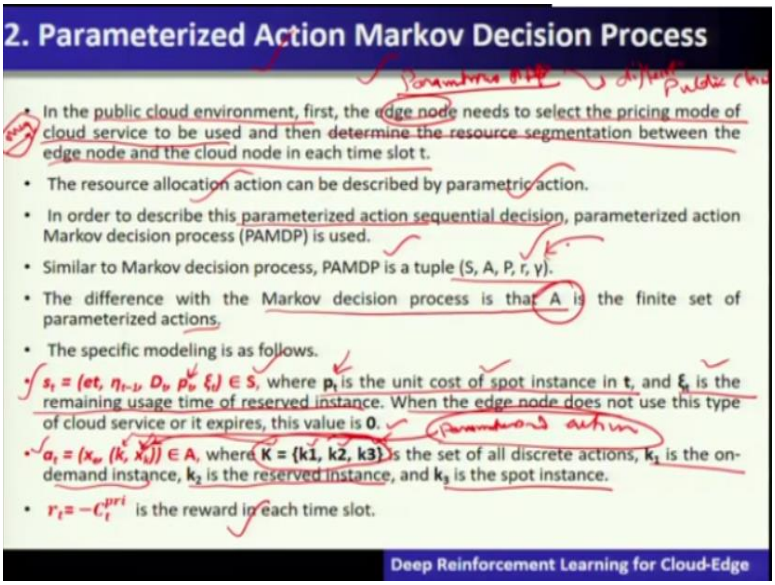
four parameters will become the part of the states in a set of finite set of states. So, it states we have defined or we have modelled.

Now, the second part is called the actions. So, out of the finance set of action, what is an action? So, action is an element of, the finite set of actions that we have considered in the Markov decision process, what an action is? So, action has two components, Xe and Xk. So, Xe is the issue of the number of virtual machines, which are provided by the edge node divided by the total number of virtual machines. So, Xe is the VMs, which are provided by the edge nodes, which is nothing but Dt of c divided by the total number of virtual machines.

So, total number of virtual machines, let us say that Dt and Xt is the ratio. X of k is the issue of the number of number of virtual machines which are provided by the cloud node to the total number of virtual machines. So, this I have explained. Then finally, the, the reward function, the, then the reward function r is the next, component here, in the Markov decision process, reward is to be represented as negative of the private cloud cost, is the reward in each private cloud cost.

That means if, let us say the demand is not met from the edge node if it is being met out of the private cloud. So, that particular minus of that cost is, so minus of that cost is called the reward.

(Refer Slide Time: 39:13)



So, let us see the, that if the edge cloud or a cloud-edge environment or a collaborative environment considers the public cloud, then that public cloud has different cost model offerings. Therefore, this particular problem of choosing which, whether on demand instance

or it is a resolved instance or a spot instance, which of these instance from that public, from a public cloud is being chosen. So, this particular decision is a discreet decision.

So, it is, now if let say public cloud also is considered in the cloud-edge environment, then this becomes a parameterized. So, parameterized action. So, Markov decision process. So, Markov decision process is parameterized, if let us say that in a public, in a different public cloud offering if it is considered. So, it is called a parameterized Markov decision process. So, let us see the problem formulation, under this particular model.

So, in a public cloud environment, first, the edge nodes needs to select the pricing mode of the cloud services to be used. And then, since it is many, out of many possibilities, it has to select and then determine the resource segmentation between the edge and cloud at this environment. So, the resource segmentation between edge and cloud environment that we have al also already seen as a Markov decision formulation. But, if you are considering different pricing model of a cloud service, then it is to be parameterized. So, therefore it is a parameterized Markov decision process. So, let us understand about this.

So, resource allocation action, because this has to be the action of a edge node. So, resource allocation action can be described by your parametric action. So, the action will become parameterized here in this case of parameterized Markov decision process. So, in order to describe this parameterized action, sequential decision, parameterized action, Markov decision processes used. Parameterized Markov decision process is again a tuple S, A P, r, gamma.

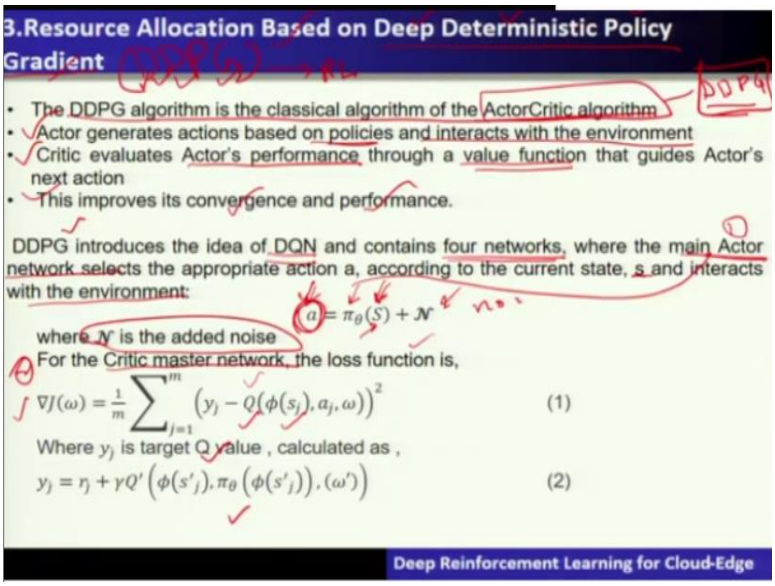
So, the difference between Markov decision process is that is a finite set of parameterized actions. So, is A parameterized action. So, the specific model is as follows. So, s is the set of states. S is an element from the capital S and s comprises of Pt is the unit cost of spot instance. And, Z is the remaining usage time of that reserve instance. So, this is the cost, and this is the time. And when the edge node does not use this type of cloud service or it expires, then it is this Pt, becomes, this Pt will become 0.

Now, action, which is parameterized, let us see how you, how it is different from the previous Markov decision problem formulation. So, action, if you recall in the previous case, it was Xe and Xk, but now this Xk that means the cloud fraction is parameterized with k. So, K is parameterized; parameterized action.

So, parameterized action in the sense the edge node has to decide, out of the different, let us say three different, type of cloud services, that is K. And this k is to be parameterized. So, where K is k1, k2, k3. k1, let us say that it is a on-demand instance, and k2 is, let us say reserved instance, and k3 is the spot instance.

Therefore, there are three type of parameters, when it comes to the cloud resources. Hence, it is called parameterized action. Whereas, the reward here reward is the minus of that particular cost function at each time slot. And gamma is the discount factor that we have considered in the previous foundation as well.

(Refer Slide Time: 44:19)



Now coming to the resource allocation strategy or problem formulation, which is not parameterized in that case, it is based on a specific reinforcement learning called deep deterministic policy gradient, DDPG - Deep Deterministic Policy and Gradient. So, DDPG. So, re resource allocation based on DDPG algorithm, we are going to explain how DDPG is being used here in the resource allocation problem formulation.

So, DDPG, let us understand about the DDPG, which is a type of reinforcement learning. So, DDPG algorithm is a classical algorithm, that is classical reinforcement learning, and it is often called as ActorCritic algorithm. So, DDPG is also called as ActorCritic algorithm. So, let us understand about a little more about this DDPG algorithm. When you say ActorCritic algorithm, actor will generate the actions based on the policies and interacts with the environment, whereas the critic will evaluate that actor's performance through a value function that guides the actor's next action.

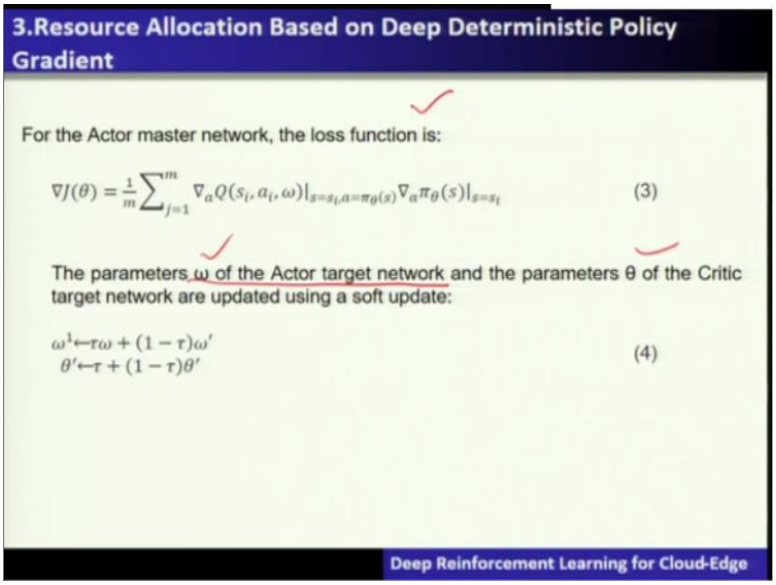
Now, this will improve its convergence and performance. So, actor and critic, both are deep networks and therefore together will improve when, when it converges, it will improve its convergence and time if, with these evaluation and action which is generated by the policy and the critic evaluates that action. And therefore, this process of DDPG goes on.

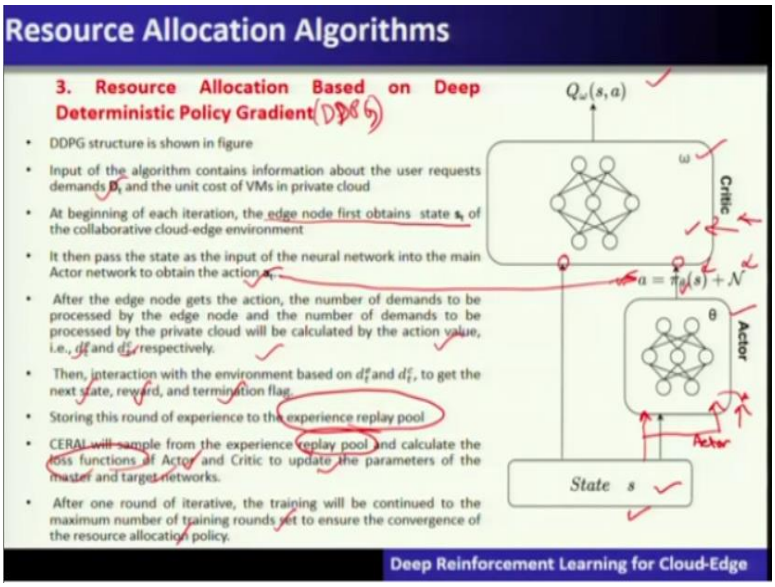
So, let us see that DDPG introduces the Deep Q networks and often DDPG contains, when you say Deep Q networks, there are four different networks is considered here in DDPG, where the main actor network, the first one is called Act Network, selects the appropriate action according to the current state and interacts with the environment. So, let us see that this is the actor network, and once the state is given, then, it will generate a particular action and is called a noise. So, noise is added over here. This is called the actor network.

So, actor network selects a particular action according to the current state. So, if, so, actor network requires the state as the input, and then, it will select particular action A. So, given a state, it will select an action, and then the critic master network will, evaluate it.

So, the second network, which will evaluate this particular actor network. So, for critic master network to evaluate, it considers a loss function. A loss function is shown where here yi is the target Q value and this is often the value which is being generated by the actor network. So, the generated Q value, both are used here to find out this loss function and this loss function is being calculated using equation 1 and 2.

(Refer Slide Time: 48:12)





So, the actor master network, if you consider the loss function is mentioned over here and the parameter omega of the actor target network. So, there are two networks is used. One is called the target network. So, these parameters of critic target networks also. So, they have the, target network, actor network and actor target network, similarly critic network and critic actor network. Finally, actor target network and critic target networks are updated and these updates are shown. Were here.

So, let us understand this entire, resource allocation, which is based on deep, deterministic policy gradient, DDPG that we have explained in the previous slide about the existence of multiple network. So, here we are showing that there are four different network. One is called Actor Network, the other is called Critic Network, then we call it as, target Actor Network and Target Critic Network. There are four different network, deep network is considered here in DDPG.

So, a state is being fed to the actor network. So, the target actor network will become a copy, of this actor network after convergence. So, when a state is fed to the actor network, so that is shown over here pi of theta. When you permit, when it takes the state with added noise, it will generate a, an action. So, it is a deep,Q network, which generates an action. This action is now given to the critic network and state also. So, there are two parameters, one is given to the credit network, which is shown by Omega.

So, this particular critic network will evaluate this particular action, which is taken by the actor network using a loss function. And, this particular loss function usin, gradient decent algorithm is quite used for, and is being, finally using the loss function it is trained to a good level. And finally, it will generate the target actor network.

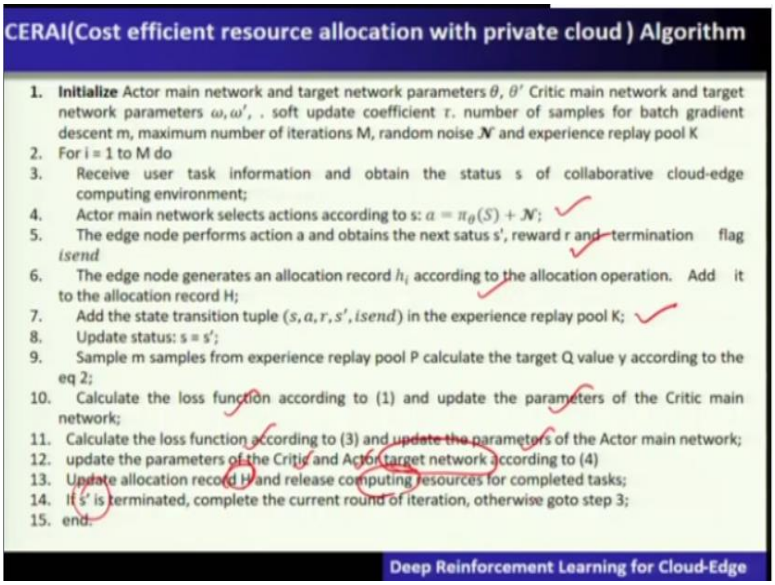
So, let us understand this particular figure we have explained. So, the input of this algorithm contains the information about the user demands, So, all these things we have already explained. So, at the beginning of each iteration, the as note first obtains the state st here, and then it passes the state as the input to the, of the neural network into the mean actor network to obtain this particular action I have already explained.

So, after the edge node gets the action, so the number of demands to be processed by the as notes and the number of demand to be processed by the private cloud will be calculated by this action value, that is, these two values, which is we have explained in previous sections, is being calculated here by the actor network.

Now, then this particular action will put on the interaction with the environment based on these values to get the next state, reward and the termination flag. So, storing this round of experience, that is into the replay pool. And then, this particular algorithm will sample, from the replay pool and calculate the loss function of the actor, so the critic, to update the parameters of the master and the target network.

So, after one round of, it, the training will continue to the maximum number of training rounds to set to convergence of the resource allocation policy. So, this is shown where here with the Q values out from this target master and the target networks.

(Refer Slide Time: 52:33)

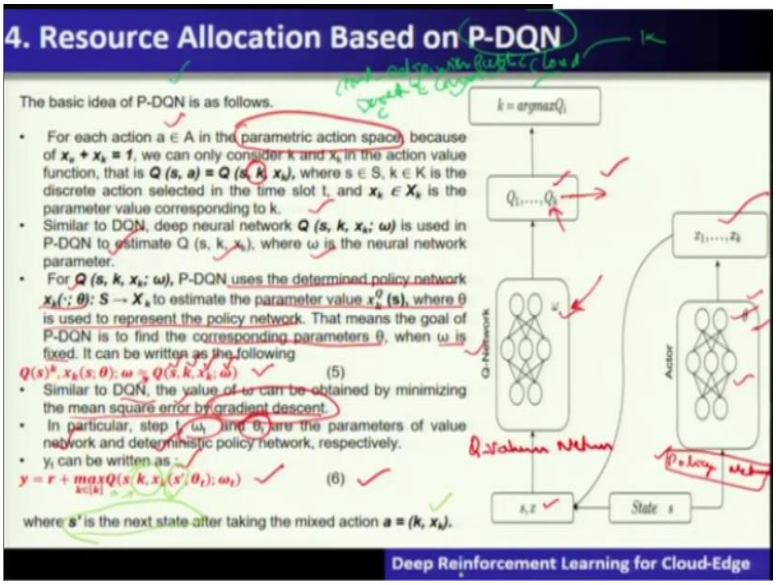


So, all the steps of this algorithm is explained over here. So, we have already explained all the steps here in this particular pool, and then calculate the loss function and then update the parameters of the critic main network, calculate the loss function, and update the parameter of

a critic of the action main network. So, there are four different networks will be formed here in this case.

So, the update, the parameter of a critic and the actor target network is done. And then update the allocation record, you know, that is to be done at least the computing resources. If s prime is terminated, complete the current round of iteration. Otherwise, it goes to step number 3 again and again, that is all explained here in this approach.

(Refer Slide Time: 53:22)



Now we are going to, discuss another resource allocation strategy, which is based on the parameterized DQN and P-DQN. So, let us understand that, in what situation this particular resource allocation is going to use the P-DQN or parameterized DQN. When the cloud in a cloud-edge environment with the public cloud. If you consider this model with the public cloud, then what we have seen in the previous slides that public cloud has offering in different types of offering that is called k, let us say the k different types. So, it is parameterized there in that case.

So, first, the edge node has to decide which of that public cloud offering has to be considered and then after that, it will be segmented based on that particular type of public cloud services. Therefore, this collaborative edge cloud environment with the public cloud often becomes the mixed, that is the discrete and continuous status space. So, with both discrete and continuous state space, we have to use a different model that is called P-DQN, a parameterized DQN, why? Because DDPG considered only the continuous state space.

So, going forward, let us understand resource allocation based P-DQN formulation. So, here, what you can see here is that in that problem formulation, this particular action is space that is called action space is parameterized. That is called parameterized action space. So, and also we know that Xc plus Xk is equal to 1. That means we can only consider the K and Xk in the action value function.

So, action value function, here can see that, Q contains s and a and instead of s and a, our two function will contains state and parameter k and Xk, where k is an element of different parameters, that means a discrete action is selected time slot, and this particular allocation that is Xk is based on that parameter value that is corresponding to k. That is what we have to now consider that what is meaning by the parameterized action is space.

So, having understood that parameterized action is space, that means the value of k over here. So, having understood that parameterized action space, now we are going to see that this DQN, similar to DQN the deep network. So, the deep neural network is used here in the Q values. So, so is used in, P-DQN also to estimate the Q values, where Omega is the neural network parameter. So, here you can see that omega is the neural network parameter here.

Now, for Q values, P-DQN is uses, determine, uses the determined policy network. So, where is the policy network? This is the policy network. So, this is the policy network. This is very important to understand. This is the policy network. So, this particular Q values to determine these Q values, P-DQN uses the policy network. So, how this Q is determined, it is determined using the policy network, and this is Xk and this particular theta. So, this you shown were here. So, this particular policy network is also, a deep Q network based policy network.

So, it determine to estimate the parameters where theta is used to represent the policy network. So, that we have already explained. This means that the goal of P-DQN is to find a corresponding parameter that is a theta when omega is fixed. So, when you fix the omega, then you have to find out which of these theta that means, which of these policy, in that policy network, will be the better one ane when omega is fixed. And that can be written over here, the Q values, on a particular omegas fixed, what will be the best policy?

So, that will be learned by these policy networks. So, these are values of Xk, k and Xk, both are needed and on fixing omega, given any state that particular Q is determined in this formulation. Now, similar to DQN, the values of Omega. Now, how this omega values you

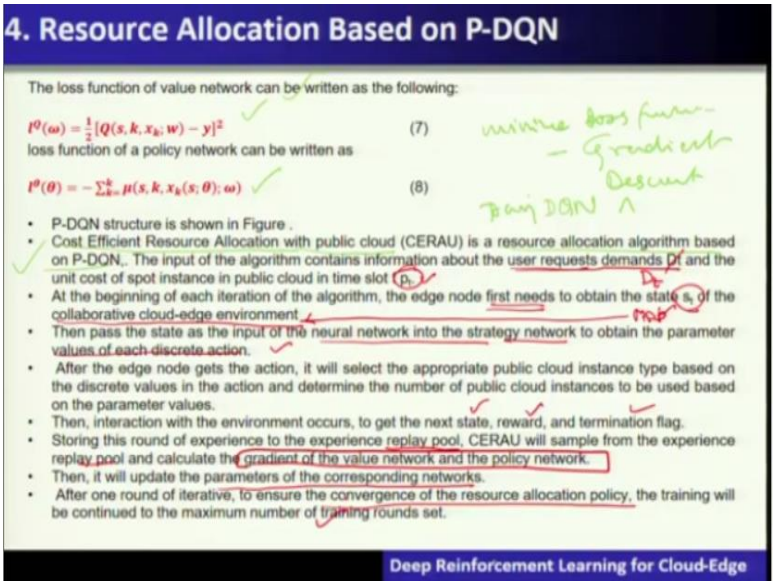
are going to fix the value of omega can be determined by minimizing the mean square by gradient descent.

So, here, this particular network will learn the best values of Omega by computing the error and then finding out the mean squared error, minimization with the help of an algorithm called gradient decent algorithm. So, all these things are already explained in the machine learning forces. So, if you are interested to know more about these terms, you have to refer to one of these machine learning courses.

Now, let us understand, given that w, find out, how to minimize that mean squared error. That means, to find out the best omega value that is the best of that Q values. So, this particular network called, Q network will formulate. So, there are two types of network. One is called, the Q network, which is represented by Omega. The other is called policy network, which is represented by the theta. So, omega T and theta T are the parameter, the values, of the value network and deterministic policy network. So, this is the policy network, and this is called a value network. So, normally it is called a Q value network sometimes.

So, there are two types of networks which are trained here in this case. And starting with the state space, the policies, using the policy, it will formulate an act. So, this particular, discrete values of or X will be determined. And then values of s and x is given to this particular value network. And value network will reinforce, find out the best of this Q value, and that can, the entire functionality of P-DQN can be returned by this public equation number 6. So, here you can see that, s prime is the next state after taking the mixed action a, which comprises of on K and Xk. So, this particular, will have the implication of an mixed action.

(Refer Slide Time: 61:34)



So, having understood this part, let us see that the loss function of the value network, how it can be written. So, this loss function of the value network is shown where here, and the loss function of the policy network is also shown over here. These loss functions are then, using the gradient decent algorithm to minimize these errors, to minimize these loss functions. So, gradient decent algorithm is used for the training deep Q networks and policy network.

So, let us understand again that the cost efficient resource allocation with the public cloud is nothing but a resource allocation algorithm, which is based on P-DQN that we have already covered. So, the input of the algorithm, here contains the user request demands Dt, and, so this, you can write like this and, the unit cost of the spot instance in a public, cloud is written by that Pt that we have already explained.

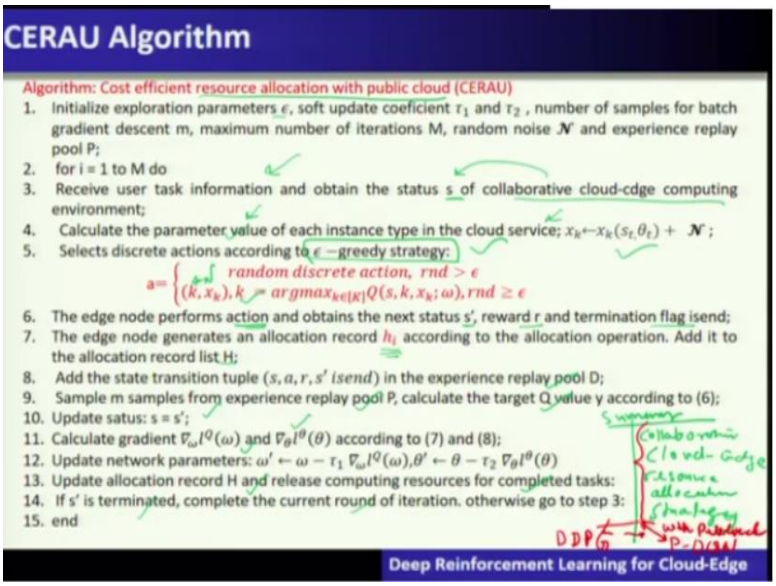
So, at the beginning of each iteration of the algorithm, the edge note first needs to obtain the state st of the collaborative edge environment. So, this is very important to understand that this collaborative edge environment will become the environment of reinforcement learning. And the state will represent this collaborative, cloud-edge environment in MDP formulation.

So, this refers to the MDP formulation and then state is given to interact with the environment and then pass the state as the input to the neural network into the strategy network to obtain the parameter values of each discrete action. So, at the edge node, after the edge node gets this action, it will then select the appropriate public cloud instance type based on the discrete values and in the action, and determine the number of public cloud instances to be based on these parameter values.

So, this is quite a straightforward in the sense that, these are some of the things which requires, the parameters. So, this is the parameter, and then the interaction with the environment occurs to get the next state, reward and the termination flag.

Now storing this round of experience in the, in the replay pool, the algorithm will sample from this replay pool and calculate the gradient of the value network and the parameter and the policy network. It will then update the parameters of the corresponding network. And after one round of arbitration to ensure the convergence of the resource allocation policy, the training will be continued to the maximum number of rounds.

(Refer Slide Time: 65:12)



So, this is all explained here in this algorithm, which we have explained it summarized again, we can, then, see this particular summarization. So, cost efficient resource allocation with the public cloud. So, that means with the public cloud, we are going to consider here. So, this algorithm uses P-DQN for that.

So, let us see that it will initialize, all these parameters and it will receive the user information, obtain the states as of a collaborative environment. So, the collaborative environment is represented with the help of a state, and it will also has now at this moment in the start point of the algorithm, the user task information that is, the demand. Now then this particular algorithm will calculate the parameter value of each instance of in the cloud service, using this particular equation that we have already explained and, selects the discrete action according to the epsilon greedy strategy.

So, this is explained over here that action is parameterized here. This is the parameter. This action is parameterized. And then, this parameter is selected using epsilon greedy strategy. Now, edge node then perform the action and obtains the next state s prime reward R and the termination flag.

So, the edge node then generates the allocation record according to the allocation operation and adds to the allocation record list H, capital H. at the state transition tuple, that is in the experience replay pool, sample, m samples from the experience pool, calculate the target Q values, and it will have now then, calculate this gradients out of that equation, which we have explained.

We will update this network parameter of both Q networks and the policy network, and we will update this allocation record, release the computing resource. If terminated, if s prime is terminated, complete the round of iteration. And then otherwise go on. So, with this, let us summarize before we, close it. So, what we have seen in the summary.

We have seen the collaborative cloud-edge resource allocation strategy. So, we have seen that if you are using with the public cloud, then this particular problem is called parameterized problem. Why because, different type of public cloud offerings are available.

So, therefore P-DQN is used in this particular case. If, let us say it is simplified, it is not using the public cloud, it is using private cloud, then in that case we can use a simply simplistic model, DDPG, that these are the two different reinforcement learning techniques which are required here in this particular use case that we have explained. So, the next class we are going to explain this and algorithm operation, taking an example, thank you.