International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 10, Issue 2, No 4, 2022. ISSN: 2348-6600
Reference ID: IJCS-440 PAGE NO: 2961-2969

CLOUD SECURITY BASED CODED CACHING WITH LARGE NUMBER OF DISTRIBUTED NETWORK

R.Sathya [#1], G.Abinaya [*2]

[#] *Master of Engineering in Computer Science & Engineering,*
A.R.J College of Engineering and Technology, Mannargudi.
[1] *sathyarengaraj028@gmail.com*

[*] *Assistant Professor, Department of Computer Science & Engineering,*
A.R.J College of Engineering and Technology, Mannargudi.

Abstract— The information theoretic models of coded caching in this paper incorporate the notion of cache-tapping. In the presence of several receivers with fixed-size cache memories and an opponent who selects symbols to tap into from cache placement and/or delivery, the wire-tap channel II is presented. The actual terminals are unaware of their locations or whether they are being tapped for placement, distribution, or both. Only the tapped set's overall size is known. For two receivers and two files, the strong secrecy capacity—the highest practicable file rate while maintaining the strong security of the whole library—is determined. Lower and higher limits on the strong secrecy file rate are determined when the library has more than two files. A code design that includes wiretap coding, security embedding codes, one-time pad keys, and coded caching is necessary for attainability. By using a genie-assisted upper bound, in which the transmitter is provided with user demands before to placement, the opposite for the two-files case is created. The upper bound is constructed using three sequential channel modifications when there are more than two files. In a cache-aided system, our findings offer verifiable security assurances against a powerful attacker that maximises its tapping during both communication stages.

Index Terms—Cloud Computing, Caching, Distributed Network.

I. INTRODUCTION

Broadband services and applications, such as holographic gaming in high definition, 8K/16K ultra-high-definition video, and immersive panoramic VR video, have significantly increased the bandwidth demand on wireless networks in recent years. For instance, 16K H. 265 full immersive VR footage demands more than 4 Gbit/s of bandwidth. The massive bandwidth burden that today's mobile network faces has prompted wireless academics and mobile operators to look for solutions to the ever-rising bandwidth demand. By using the mobile device's caching size, the coded caching strategy is suggested in the seminal work as a way to reduce traffic congestion. In the meanwhile, authors discover that in large-scale wireless heterogeneous networks, utilising the caching resources at the relay and consumers may significantly increase throughput. When we examine the wireless spectrum bottleneck issue, we find that internet apps and services heavily rely on asynchronous content reuse, which makes them very different from more traditional services like phone calls and text messaging. Thus, by caching

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 10, Issue 2, No 4, 2022. ISSN: 2348-6600
Reference ID: IJCS-440 PAGE NO: 2961-2969

popular material at the wireless edge during off-peak hours, bandwidth may be saved while peak-hour network congestion is reduced.

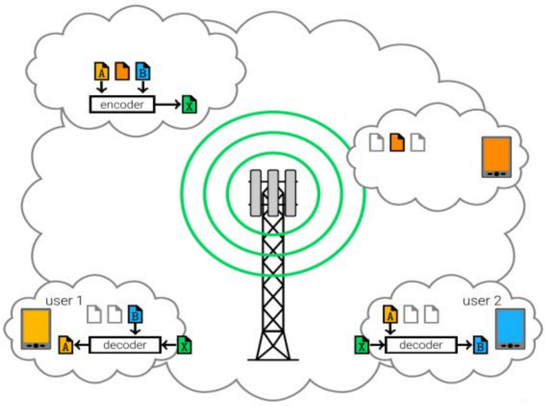

Fig.1 Step by Step file Transfer in Cloud

The research mentioned above show that edge caching has been viewed as a critical enabling technology in future wireless networks to address the wireless spectrum crunch issue, for example, at a base station (BS) or the user devices. As a consequence, in contrast to our earlier work and the standard work on the coded caching, the coded cache with device computing has greater space thanks to the collaboration of edge caching and computation at the mobile device.

RELATED WORK

Numerous publications on the topic of coded caching generate a new lower bound on the rate of transmission of any coded caching technique while taking into account an arbitrary popularity distribution. The new data inner bound on the memory-rate tradeoff is presented by the authors after they offer a fresh coded method without user coordination that demonstrates the relationship between the uncoded and coded prefetching schemes. The coding schemes and information theoretic constraints for a D2D caching network have been identified in this research. It should be noted that the studies mentioned above on the coded caching approach solely evaluate user storage without taking device computing into account. However, there are a number of further publications regarding wireless content caching. The BS density can be decreased by raising the BS cache size when it exceeds a certain threshold. When those articles are taken into account within the context of the classic MEC design, we see that computational jobs are produced at user devices on both counts. Several articles additionally combine the conventional MEC design with immersive VR applications. Optimises the viewport rendering offloading approach using the MEC server's and VR devices' computational power. For proactive computing and frame caching, the study makes use of the cache-aided and computing-aided edge server. In order to calculate VR input data products in user devices, the edge computing server is utilised. However, in the aforementioned studies, the user device's cache capacity was not taken into account. Therefore, to reduce the transmission bandwidth in the MEC system, our earlier work fully utilises edge computing and caching to simultaneously optimise the cache and computation strategy on user devices. In contrast to the classic MEC design, the MEC server now generates computing jobs like video on demand. This allows the BS in those publications to offload the data to user devices, which then do the partial rendering to reduce the cost of bandwidth. The coded caching approach, however, has not been taken into account by earlier researchers.

LITERATURE SURVEY

Benefits of Edge Caching With Coded Placement for Asymmetric Networks and

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 10, Issue 2, No 4, 2022. ISSN: 2348-6600
Reference ID: IJCS-440 PAGE NO: 2961-2969

Shared Caches: Abdelrahman M. Ibrahim, Ahmed A. Zewail, Aylin Yener_2021

This study examines a network with cache support in which users can access helper caches of various sizes. First, it is suggested to employ coded placement strategies to take advantage of the differences in cache sizes that occur when just one user is connected to each cache. The unicast/multicast signals designed to support users connected to tiny memories are used in the proposed approach to decode the contents of the bigger memories. For three-user systems with arbitrary cache sizes, bigger systems, and systems operating in the small total memory regime, a reduction in delivery load with coded placement is demonstrated when compared to uncoded placement.

Decentralized Coded Caching for Shared Caches: Monolina Dutta, Anoop Thomas_2021

The client-server framework's client requests display temporal variability, which causes network congestion at unpredictable intervals. Popular material is placed into cache memory dispersed throughout the network to address this issue. Each user has an associated cache in the traditional cache system, and cache loading is centralised. A more feasible strategy for big networks is to decentralise cache loading. The shared caching dilemma, in which one cache can serve several customers, is discussed in this letter. The decentralised shared caching problem is addressed by the suggestion of a fresh and ideal delivery strategy.

Multi-source spinal coding for coded caching multicast transmissions in wireless networks: Aimin Tang, Xudong Wang_2021

Coded caching has recently been viewed as a viable method to lessen the traffic load in wireless networks. The time-varying heterogeneous channel conditions must be taken into consideration in order to provide highly effective coded caching multicast broadcasts. A useful and innovative multi-source spinal coding (MSSC) method is created in this study for coded caching multicast broadcasts in heterogeneous channel settings. For various users, MSSC can accomplish uneven link rates in multicast transmissions by investigating combined network coding and spinal coding (SC) designs.

Cache-Aided Polar Coding: From Theory to Implementation: Yasser Fadlallah, Othmane Oubejja, Sarah Kamel_2021

In this research, we offer an extended coded caching strategy for add multiple networks with decentralised caching, based on piggyback coding. The original coded caching method, which is based on index encoding and an information assignment that may be carried out with minimal graph-coloring, is expanded to create the suggested scheme by adopting Polar codes. The coded caching is modified to cater for varying user coding rates and to mix transmissions to cache-aided and cache-free users. Polar codes are modified to allow users to apply portions of their cache contents as the "frozen bits" for Polar decoding.

A Coded Caching Scheme with Linear Sub-packetization and its Application to Multi-Access Coded Caching: Anjana Ambika Mahesh, B. Sundar Rajan_2021

This study presents a new coded caching strategy that is inspired by the symmetric neighbouring consecutive side information index coding problem, which tackles the issue of exponentially rising thread with the number of users in a centralised coded caching system. With

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 10, Issue 2, No 4, 2022. ISSN: 2348-6600
Reference ID: IJCS-440 PAGE NO: 2961-2969

no limitations on user count or file size, the scheme has a placing policy where the amount of sub-packets needed rises only linearly with user count and an instantly decodable delivery policy.

METHODOLOGY

Approach:

In order to prevent local optimization and stagnation and to guarantee learning rates of different dimensions for each ant and ant, the DCAA algorithm maximises the multi-population strategic plan, the neighbourhood detailed learning method, the fine research design, the chaotic optimization strategy, the super outstanding ant strategy, the disciplinary strategy, and the min-max ant strategy. The goal of the study is to develop the DCAA algorithm. In order to prevent local optimization and stagnation and to guarantee learning rates of various dimensions for each ant and ant population, the DCAA algorithm fully utilises this same multi-population strategy, neighbourhood clear learning strategy, fine search strategy, chaotic optimization strategy, super excellent ant strategy, punishment strategy, and min-max ant strategy. Benchmark functions and dynamically changing PID parameter settings are analysed to show the efficacy of the DCAA method.

Algorithm:

Delta Cube Access Algorithm (DCAA)
DCAA(E′, p) → cp
PMquirN: ROT has M’ and л ;ORP has sk; KNN
Step 1: KNN:
(a). ComputN Mpk(qj), for 1 ≤ j ≤ m
(b). KNnd Mpk(q) = < Mpk(qi), …, Mpk(qm) > to C1
Step 2: ROT and ORP:
(a).ROTrNcNivNs Mpk(q) from KNN
(b). for i = 1 to n do:
β′i ← Dsk(βi), for 1≤ i ≤ n
- if β′i = 0, thNn U′i = Mpk(1)
(c).ORP: Mpk(di) ← SSND (Mpk(q), Mpk(ti))
(d).ROT: V ← л-1(U′)
Mpk(di, γ) ← SBOR (Vi, Mpk(di, γ))

Step 3: for p = 1 to k do:
(a).ROT and ORP:
([dmin]), Mpk(I), Mpk(c′) ← SMINn(ɵ1, . . . , ɵn), whNrN
ɵi = ([di], Mpk(Iti), Mpk(ti, m+1))
Mpk(c′s) ← Mpk(c′)
(b). C1:
▲ ← Mpk(I)N-1
for i = 1 to n do:
Step 4: KCNCk(Mpk(c′1), . . . , Mpk(c′k))

Architectural View:

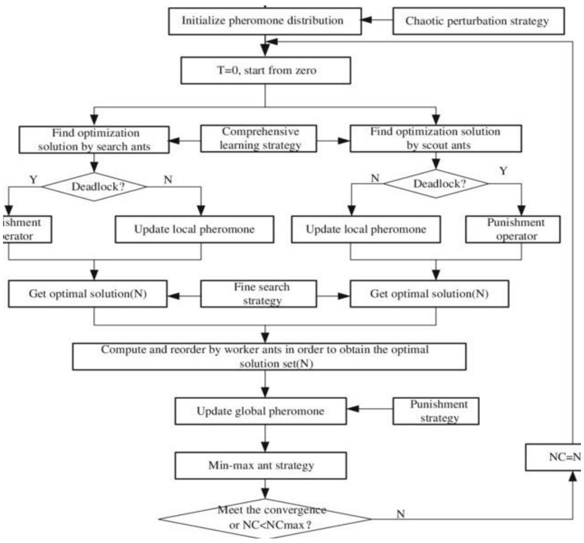

Fig.2 Delta Cube Access Approach

Approach:

The fundamental paradigm of the Heterogeneous Based Dynamic Allocation (HBDA) algorithm is enhanced by the Heterogeneous Based Dynamic Algorithm. The experiment's findings show that the

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 10, Issue 2, No 4, 2022. ISSN: 2348-6600
Reference ID: IJCS-440 PAGE NO: 2961-2969

algorithm is capable of resolving the issue of work scheduling in the cloud.

Future scheduling decisions that aim to minimise processing time and lighten the load on data centres may be based on the findings of this research. It can also be applied to other policies, and additional metrics can be used to create a policy that is more optimal. The HBDA algorithm has been successfully used to solve numerous NP-hard optimization issues. One of the first problems to which HBDA was effectively applied was the travelling salesman problem (TSP). The HBDA algorithm employs a large number of cooperative agents, or "artificial ants," to search for the best answer.

Algorithm:

Heterogeneous Based Dynamic Allocation Algorithm

Input: graph G = (V,E)

P = ∅ ;

While U is not empty do

Get some vertex v ∈ U

If P V {v} is a clique then

R = P V {v}

End if

Remove v from U

End while

Return R

Architectural View:

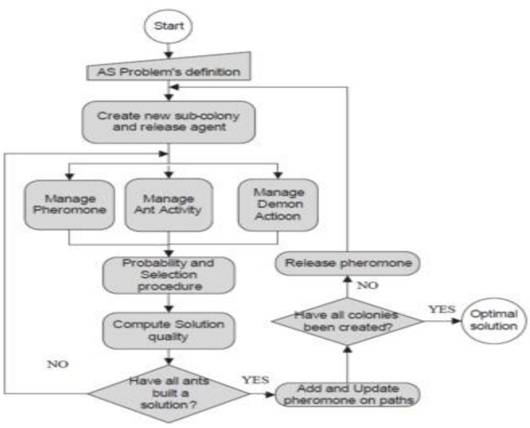

Fig.3 Heterogeneous Based Dynamic Allocation Approach

Approach:

ACAA is a stochastic search algorithm for the appropriating of limited natural resources that is based on the concepts of inter-individual natural competition. The genes of the winner usually determine how successful they are, and those people's procreation allows their genes to spread. The population would gradually acquire more natural resources through the selection of superior people and their reproduction. The simulation of this process using the ACAA leads to the selection of the best objective functions. The typical ACAA is often not practical for solving complex situations. The method works with string structures, which resemble biological structures in that they evolve over time according to the survival of the fittest throughout a randomised yet well-organized exchange of information. Since the principles of competitive forces are the foundation of the stochastic search algorithm ACAA, a new set of strings is created after each generation by using pieces of the most suitable individuals from the previous set. These strings will include the general characteristics and affecting elements of the problem. Each string represents a potential network configuration. The objective function, or the value of the power and service consumption, as well as the number of units to be used in the modelled network design, or the total capital and operating cost, will be determined for each of the strings.

Algorithm: Access Cloudlet Array Algorithm (ACAA)

Output: Perfect Decision Making for User Query

Step 1: GET a set of doc P for a target event.

Step 2: APPLY the classification to obtain value vt = {0, 1}.

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
SKRGC Since 2012
http://www.ijcsjournal.com Volume 10, Issue 2, No 4, 2022. ISSN: 2348-6600
Reference ID: IJCS-440 PAGE NO: 2961-2969

Step 3: FOR EACH tweet tw ∈ TW, obtain features P, Q and R.

Step 4: CALCULATE event occurrence probability p occur using vt, t ∈T;

Step 5: PUT a doc D using search API every s seconds and obtain tweets TW.

Step 6: user Ut. Set lt = null if both do not work.

Step 7: IF it is above the threshold reoccur, then proceed to step 5.

Architectural View:

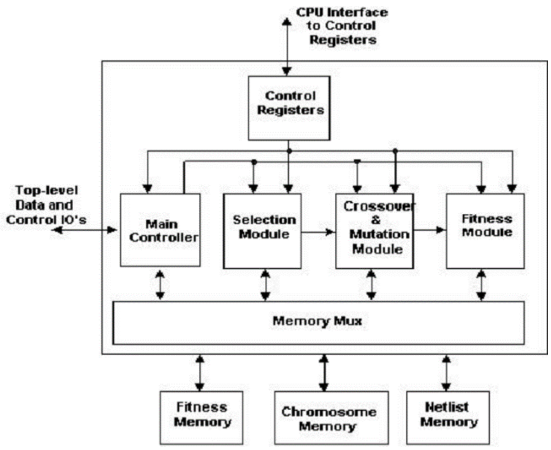

Fig.4 Access Cloudlet Array Approach

Proposed Approach:

Finding the most effective answer by using auction theory MCN's High Security Distributed Algorithm(HSDA). The overloaded gateway in HSDA detects the problematic gateways and disallows their participation in the auction. The challenge and several techniques to solving it in an MCN setting are defined theoretically. Finally, we are examining the proposed scheme's Nash equilibrium. In addition to doing a theoretical study, we expand the technique to the scenario of numerous users. The correctness of the suggested algorithms is evaluated numerically. The following advantages should be preserved by a user authentication process for mobile users in a distributed cloud service environment. The authentication technique aims to demonstrate global authentication and user confidentiality without using SSL or any other reliable cryptosystems. While involvement in the user login session is optional, user registration and service provider registration need the assistance of a reliable third party. A user can access the mobile services of several service providers with just one encryption key. The authentication method does not necessitate complicated computations on user's mobile devices.

Algorithm: High Security Distributed Algorithm (HSDA)
Inputs: Nil(b(Qil)ˆ;Pil)
Output: Qami*
1: Invoke High Security Distributed Algorithm for single user share
2: elseif user not assign then
3: else
4: elseif Pi(H + 1) -Qiniti < 0 then
5: End
6: Bandwidth Distribution required.
7: elseif (min(ej; de)≤ 1then
8: Hi pays #i to the assignee gateway
9: Assign the user to Qami
10: Hi shares the bid p with the participant
11: else
12: elseif any Qami_ accepts the price then
13: Assign the user to Qami with minimum de i
14: elseif multiple gateways accept the price then
15: else
16: Revise the bid price q(M + 1) = p(M) + Δp
17: Goto Step 2
18: End
19: End
20: End

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
Since 2012
http://www.ijcsjournal.com Volume 10, Issue 2, No 4, 2022. ISSN: 2348-6600
Reference ID: IJCS-440 PAGE NO: 2961-2969

Architectural View:

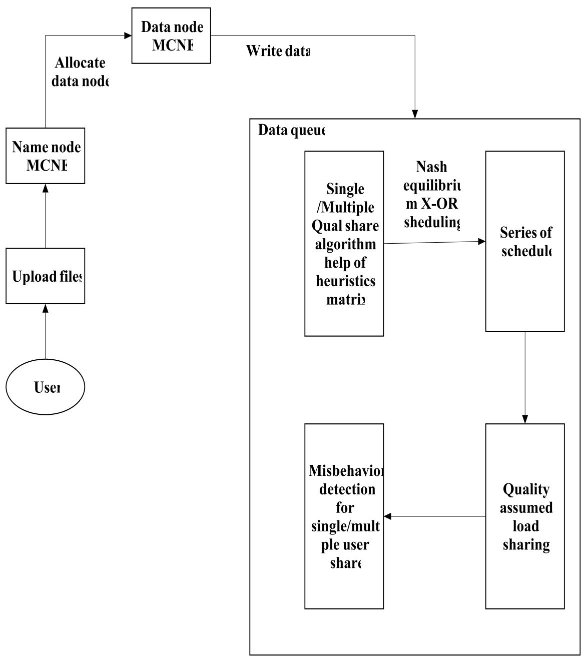

Fig.5 High Security Distributed Approach

EXPERIMENTAL RESULT

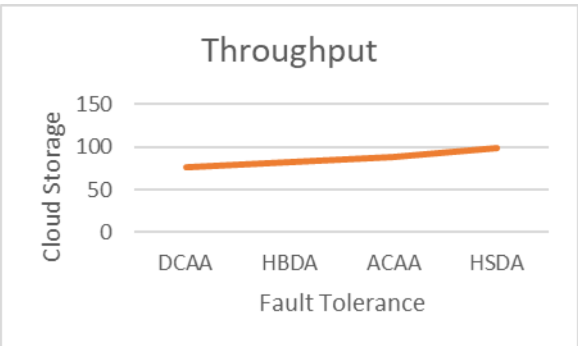

Fig.6 Throughput

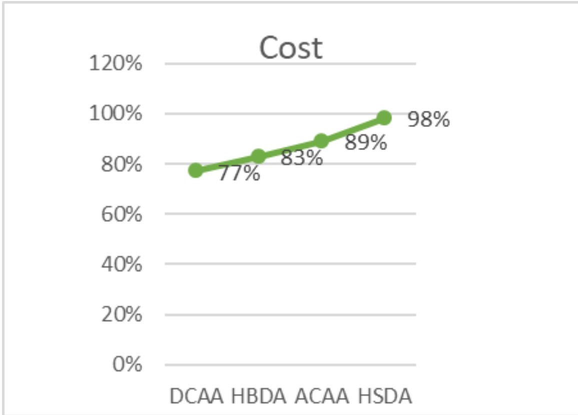

Fig.7 Cost

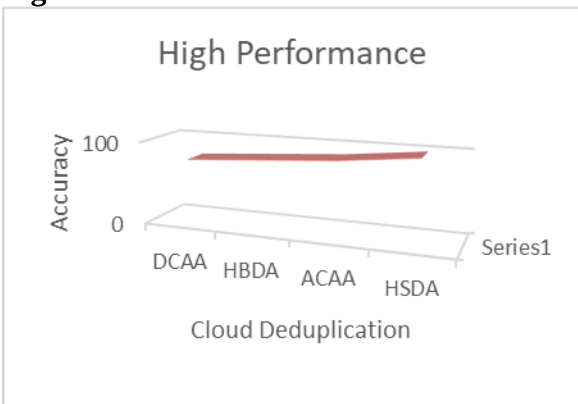

Fig.8 High Performance

CONCLUSION

By utilising the computational power and caching capabilities of hardware equipped with the coded cache technique in the MEC system, we have investigated the issue of how to reduce the average transmission bandwidth. Then, a coded caching with device computing technique is suggested to reduce the average bandwidth while accounting for cache size, computation job time, and average device energy consumption. When the number of devices and processing workloads increase, the defined issue is a large-scale combination of nonconvex and smooth integer programming. Since it is obvious that the programming in the article is challenging to tackle, we have divided it into a number of smaller challenges that can each be handled effectively on their own. The numerical outcomes demonstrate that the three state-of-the-art benchmarks are greatly outperformed by the coded cache with devices computing technique. We did not take into account the spatiotemporal relating to money in our work. The approach might be expanded to incorporate a device computing technique for the spatiotemporal important for marketing.

REFERENCES

1. Beamforming Design for Wireless Coded Caching with Different Cache Sizes: Antti Tölli_2022
2. Rajkumar, V., and V. Maniraj. "Hybrid Traffic Allocation Using Application-Aware Allocation Of Resources In Cellular Networks." Shodhsamhita (ISSN: 2277-7067) 12.8 (2021).
3. Linear Network Coded Wireless Caching in Cloud Radio Access Network: Long Shi _2021
4. Secretive Coded Caching With Shared Caches: Shreya Shrestha Meel_2021
5. Coded Caching Does Not Generally Benefit From Selfish Caching: Federico _2022
6. On the Fundamental Limits of Fog-RAN Cache-Aided Networks With Downlink and Sidelink Communications: Kai Wan_2021
7. Exploiting Joint-Cache-Channel Coding for Decentralized Coded Caching With Heterogeneous Link Rates and Cache Sizes: Aimin Tang, Yao Liu, Xudong Wang_2021
8. Rajkumar, V., and V. Maniraj. "RL-ROUTING: a deep reinforcement learning sdn routing algorithm." journal of education: rabindrabharati university (ISSN: 0972-7175) 24.12 (2021).
9. A Novel Transformation Approach of Shared-link Coded Caching Schemes for Multiaccess Networks: Giuseppe Caire_2021
10. "Dependency Aware Caching (Dac) For Software Defined Networks.", Rajkumar, V., and V. Maniraj.Webology (ISSN: 1735-188X) 18.5 (2021).
11. Benefits of Edge Caching With Coded Placement for Asymmetric Networks and Shared Caches: Aylin Yener_2021
12. Rajkumar, V., and V. Maniraj. "Privacy-Preserving Computation With An Extended Framework And Flexible Access Control." 48.10 (2021).
13. Decentralized Coded Caching for Shared Caches: Anoop Thomas_2021
14. Coded Caching with Shared Caches from Generalized Placement Delivery Arrays: Elizabath Peter, B. Sundar Rajan_2021
15. Multi-source spinal coding for coded caching multicast transmissions in wireless networks: Aimin Tang, Xudong Wang_2021
16. "Software-Defined Networking's Study with Impact on Network Security." Design Engineering (ISSN: 0011-9342) 8 (2021), Rajkumar, V., and V. Maniraj.
17. Cache-Aided Polar Coding: From Theory to Implementation: Sarah Kamel_2021
18. ECC-United Cache: Maximizing Efficiency of Error Detection/Correction Codes in Associative Cache Memories: Soontae Kim_2021

http://www.ijcsjournal.com
Reference ID: IJCS-440
Volume 10, Issue 2, No 4, 2022.
ISSN: 2348-6600
PAGE NO: 2961-2969

19. Rajkumar, V., and V. Maniraj. "HCCLBA: Hop-By-Hop Consumption Conscious Load Balancing Architecture Using Programmable Data Planes." Webology (ISSN: 1735-188X) 18.2 (2021).
20. A Coded Caching Scheme with Linear Sub-packetization and its Application to Multi-Access Coded Caching:B. Sundar Rajan_2021