THESIS

Controller Placement and Radio Resource Allocation for D2D Communications in 5G Wireless Networks

Yarisley Peña Llerena

Brasilia, December 2022

BRASILIA UNIVERSITY

FACULTY OF TECHNOLOGY

BRASILIA UNIVERSITY
Faculty of Technology
Department of Electrical Engineering

THESIS

Controller Placement and Radio Resource Allocation for D2D Communications in 5G Wireless Networks

Yarisley Peña Llerena

Thesis submitted to the Electrical Engineering Department as a partial requirement to obtain the degree of PhD in Electrical Engineering

Examination Board

Paulo Roberto de Lira Gondim, Ph.D, FT/UnB ______________________
Orientador

Renato Mariz de Moraes, Ph.D, CIN/UFPE ______________________
Examinador externo

Luiz Carlos Pessoa Albini, Ph.D, UFPR ______________________
Examinador externo

Carlos Humberto Llanos Quintero, Ph.D, ENM/UnB ______________________
Examinador interno

FICHA CATALOGRÁFICA

LLERENA, YARISLEY PEÑA

Controller Placement and Radio Resource Allocation for D2D Communications in 5G Wireless Networks [Distrito Federal] 2022.

xvi, 153 p., 210 x 297 mm (ENE/FT/UnB, PhD, Engenharia Elétrica, 2022).

Thesis - Brasilia University , Technology Faculty.

Department of Electrical Engineering

1. 5G	2. Device-to-device
3. Controller Placement	4. Resource Allocation
5. Meta-heuristic	6. URLLC
I. ENE/FT/UnB	II. Título (série)

REFERÊNCIA BIBLIOGRÁFICA

LLERENA, Y. (2022). *Controller Placement and Radio Resource Allocation for D2D Communications in 5G Wireless Networks* . Publication PPGEE.TD 190/22. Thesis, Department of Electrical Engineering, Brasilia University, Brasília, DF, 153 p.

CESSÃO DE DIREITOS

Yarisley Peña Llerena

SQS 403 Bloco C apto 202, Asa Sul.

70237-030 Brasília - DF - Brasil

Acknowledgements

Agradeço a Deus por ter me dado a oportunidade de cumprir o sonho de ser doutora.

Ao meu orientador, pela sua paciência, pelo apoio, por ter me recebido como sua orientanda e ter me guiado no desenvolvimento do meu projeto de doutorado, Prof. Paulo muito obrigada.

Ao meu esposo por estar ao meu lado, pela sua compreensão, pela sua ajuda, pela sua confiança em mim e sobretudo pelo seu amor. À minha mãe, meus sogros, minha tia Teresa, meus primos e resto da família em Cuba por ter me apoiado neste tempo que tenho estado longe deles, mas mesmo estando longe me fizeram sentir que estavam por perto, me cuidando e me acompanhando. Às pessoas que conheci em Brasília e às que conviveram comigo no dia a dia fazendo me sentir como em família com elas.

Ao pessoal do Laboratório de Redes da UnB e ao Programa de Pós-graduação em Engenharia Elétrica da UnB. À Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), pelo apoio financeiro.

A todos os que de algum jeito se interessaram pelo sucesso deste trabalho.

Muito obrigada,

Yarisley Peña Llerena

RESUMO EXPANDIDO

Título: Localização de Controladores e Alocação de Recursos de Rádio para Comunicações D2D em Redes Sem Fio 5G
Autor: Yarisley Peña Llerena
Orientador: Prof. Dr. Paulo Roberto de Lira Gondim, FT/UnB
Programa de Pós-Graduação em Engenharia Elétrica - PPGEE

As redes celulares de próxima geração prometem uma transmissão de dados mais rápida, maior largura de banda e menor latência, e a tecnologia de comunicação Dispositivo-a-Dispositivo (D2D, do inglês *Device-to-Device*), devido aos serviços baseados em proximidade (ProSe, do inglês *Proximity Services*) dos dispositivos de comunicação, ajuda a tornar tais promessas uma realidade. As vantagens de tal comunicação incluem maior vazão, maior taxa de transmissão de dados, menor latência, menor consumo de energia, equidade e maior eficiência espectral.

Este trabalho possui foco na gestão das comunicações D2D em uma rede de comunicação móvel celular, sendo tratados dois problemas: o problema de localização de controladores (CPP, do inglês *Controller Placement Problem*) e o problema de alocação de recursos de rádio (RRA, do inglês *Radio Resource Allocation*).

A técnica proposta para resolver o problema CPP envolve, primeiramente, a obtenção do número de comunicações D2D a serem gerenciadas e depois o número de controladores necessários, a sua localização física na infra-estrutura celular e a melhor atribuição de eNB (do inglês *evolved NodeB*) aos controladores. O problema foi modelado como um problema de optimização e as meta-heurísticas Colônia de Abelhas Artificiais (ABC, do inglês *Artificial Bee Colony*) e Sistema de Colônia de Formigas com Memória Externa (ACS-EM, do inglês *Ant Colony System with External Memory*) foram utilizadas para resolvê-lo. As duas meta-heurísticas foram comparadas com as meta-heurísticas Sistema de Colônia de Formigas (ACS, do inglês *Ant Colony System*) e Optimização do Enxame de Partículas (PSO, do inglês *Particle Swarm Optimization*). A análise revelou que as complexidades computacionais do ABC e do PSO são inferiores às do ACS-EM e do ACS. No entanto, ABC e ACS-EM demonstraram um melhor desempenho na resolução do problema, com características que permitem uma exploração mais eficiente do espaço de busca, evitando assim soluções sub-ótimas. Os melhores resultados foram obtidos com o ABC, seguido pelo ACS-EM, depois pelo ACS e, por último, pelo PSO.

Com o fim de obter o número de comunicações D2D admitidas na rede, dois métodos foram propostos para a solução do problema RRA. No primeiro, o problema de optimização é resolvido pela meta-heurística *Social-Aware RRA Artificial Bee Colony* (SA-RRA-ABC), considerando a relação social entre os usuários, um esquema de *feedback*, e a maximização do *throughput* do sistema. Um esquema de *feedback*, chamado *selected-NM Maximum Distance Ratio (MDR) q-bit feedback*, foi proposto para reduzir a sobrecarga de sinalização (*overhead*). Cada receptor D2D envia à eNB apenas q-bits de informação do estado do canal de N dispositivos celulares (CUE, do inglês *Cellular User Equipment*) e de M pares D2D, utilizando na seleção o maior valor da métrica MDR. A meta-heurística SA-RRA-ABC foi validada através de simulações e comparada com os algoritmos gulosos GRAA (do inglês *Greedy Resource Allocation Algorithm*) e SA-GRAA (do inglês *Social-aware Greedy Resource Allocation Algorithm*). Os resultados mostraram que a meta-heurística SA-RRA-ABC apresenta melhor desempenho que os algoritmos gulosos. Foi também mostrado que o modelo de *feedback* proposto pode alcançar desempenho próximo do modelo *full CSI - Channel State Information* (onde a eNB tem informação do estado de todos os canais), com uma sobrecarga de sinalização (*overhead*) inferior.

O segundo método de solução do problema RRA considera o problema de optimização em uma rede celular D2D que oferece serviços URLLC (do inglês *Ultra-reliable and Low Latency Communications*), para o envio de pacotes curtos diretamente entre usuários, utilizando comunicações D2D, maximizando assim a eficiência energética da rede. O problema é resolvido por três meta-heurísticas bio-inspiradas (ABC, ACS-EM e PSO), levando em consideração a interferência quando os usuários de celulares e dispositivos D2D utilizam um mesmo recurso de rádio. As meta-heurísticas foram comparadas com uma heurística gulosa e um algoritmo de busca exaustiva. A análise revelou que a complexidade computacional da heurística gulosa é a mais baixa e que as complexidades do ABC e do PSO são mais baixas que as do ACS-EM. Porém, o ABC mostrou um melhor desempenho na resolução do problema, seguido pelo ACS-EM, depois pelo PSO e, por último, pela heurística gulosa.

Palavras-chave: 5G, Dispositivo-a-Dispositivo, Localização de Controladores, Alocação de Recursos, Meta-heurísticas, URLLC.

ABSTRACT

Next generation cellular networks promise faster data transmission, higher bandwidth, and lower latency and the Device-to-Device (D2D) communication technology helps make such promises a reality, due to Proximity Services (ProSe) of communication devices. The advantages of such communication include higher throughput, enhanced data rate, lower latency and energy consumption, fairness, and improved spectral efficiency.

This research focuses on the management of D2D communications on a cellular network. Two problems, namely Controller Placement Problem (CPP) and Radio Resource Allocation (RRA) were studied.

The technique proposed for solving the former involves the obtaining of number of D2D communications to be managed and then number of controllers required, their physical location in the cellular infrastructure, and the best evolved NodeB (eNB) assignment for them. The problem was modeled as an optimization problem one and Artificial Bee Colony (ABC) and Ant Colony System with External Memory (ACS-EM) meta-heuristics solved it. They were compared with Ant Colony System (ACS) and Particle Swarm Optimization (PSO) algorithms, and the analysis revealed the computational complexities of ABC and PSO are lower than of ACS-EM and ACS. However, ABC and ACS-EM showed better performance in solving the problem, with characteristics that enable a more efficient exploration of the search space, thus avoiding sub-optimal solutions. The best results were obtained with ABC, followed by ACS-EM, then ACS, and lastly PSO.

Regarding the RRA problem, two solution methods were proposed towards obtaining the number of D2D connections admitted in the network. In the first RRA solution method, the optimization problem is solved by the Social-aware RRA Artificial Bee Colony (SA-RRA-ABC), considering the social relationship between users, a feedback scheme, and maximization of the system throughput. A selected-*NM* Maximum Distance Ratio (MDR) q-bit feedback scheme designed reduces feedback overhead, since each D2D receiver sends only q-bit feedback Channel State Information (CSI) among N Cellular User Equipment (CUE) and M D2D pairs with the largest MDR metric. SA-RRA-ABC was validated through simulations and compared with Greedy Resource Allocation Algorithm (GRAA) and Social-aware Greedy Resource Allocation Algorithm (SA-GRAA). The simulation results showed its better performance. The selected-NM q-feedback model proposed can achieve performance close to that of the full CSI model with lower overhead.

The second RRA solution method deals with the optimization problem in a D2D cellular network that offers Ultra-reliable and Low Latency Communications (URLLC) services for the sending short packets directly to their destination, thus maximizing the network energy efficiency. The problem is solved by three bioinspired algorithms, namely ABC, ACS-EM, and PSO, which take into account interference when cellular and D2D users use a same radio resource. The meta-

heuristics were compared with a greedy heuristic and an exhaustive search algorithm and the analysis revealed the computational complexity of Greedy is the lowest and those of ABC and PSO are lower than that of ACS-EM. However, ABC showed better performance in solving the problem, followed by ACS-EM, then PSO, and lastly Greedy heuristic.

SUMMARY

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

Latin Symbols

a_c	Particle Swarm Optimization (PSO) algorithm parameter - a coefficient
A_p	PSO algorithm parameter - set with possibilities
$B = \{b_k\} = \{b_1, b_2, \ldots, b_b\}$	set of eNBs
b	cardinality of set B
b_{c_i}	number of eNBs managed by SDN controller c_i
$Clinks$	SA-RRA-ABC algorithm variable - set of subchannels
$C = \{c_j\} = \{c_1, c_2, \ldots, c_c\}$	set of SDN controllers
c	cardinality of set C
$c1$	PSO algorithm parameters - individual-cognition
$c2$	PSO algorithm parameters
C_c	CPP - cost function associated with the number of controllers
C_{D2D}	the number of D2D pairs allowed in each eNB
$D = \{d_j\} = \{d_1, d_2, \ldots, d_d\}$	set of D2D pairs
d	cardinality of set D
$dist_{ij}$	distance between node i and node j
d_j^R	DUE receiver
d_j^T	DUE transmitter
E	PSO algorithm parameter - crisp set
e	PSO algorithm parameter - crisp set element
f	PSO algorithm parameter - number of particles
$f(x)$	the objective function that must be minimized or maximized along with the decision variables
$g_{Tx,Rx}$	the channel gain from transmitter Tx to receiver Rx
$g_{d_j^T,d_j^R}$	the channel gain from transmitter d_j^T to receiver d_j^R
$g_{d_j^T,b_k}$	the channel gain from d_j^T to eNB b_k
g_{u_i,b_x}	the channel gain from u_i to eNB b_k
$gbest_j$	PSO algorithm parameter - the particles share information on the best point achieved
$G_C(V_{tx}, \Lambda)$	construction graph
$I_{Tx,Rx}$	the interference received at Rx from Tx
i	index
$I_{\triangle u_i,b_k}$	interference channel gains from d_j^T to eNB b_k
I_{u_i,d_j^R}	the interference received at d_j^R from u_i

I_{d',d_j^R}	the interference received at d_j^R from D2D pair transmitters using a same resource blocks (RBs) as d_j
$I_{d_j}^{max}$	maximum tolerable interference in D2D link d_j
$I_{u_i}^{max}$	maximum tolerable interference in the communication between u_i and eNB
$IR_{d_{j_1}^R}$	a potentially larger interference range of receptor $d_{j_1}^R$
$IR_{d_{j_2}^R}$	a potentially larger interference range of transmitter $d_{j_2}^R$
IR	a potentially larger interference range
j	index
j_f	CPP - Jain's index
J_f	CPP - cost function associated with Jain's index
K	SDN controller capacity
k	index
l	distance
$l_{d_{j_1}^T,d_{j_2}^R}$	distance between D2D $d_{j_1^T}$ and D2D $d_{j_2}^R$
$l_{d_{j_2}^T,d_{j_1}^R}$	distance between D2D $d_{j_2^T}$ and D2D $d_{j_1}^R$
$l_{d_{j_1},d_{j_2}}$	distance between D2D d_{j_1} and D2D d_{j_2}
$L_{Tx,Rx}$	path-loss model from transmitter Tx to receiver Rx
l_{u_i,b_k}	distance between CUE u_i and eNB b_k
l_{u_i,d_j^R}	distance between CUE u_i and d_j^R
$L_{d_j^T,d_j^R}$	path-loss model from d_j^T to d_j^R
$limit$	ABC algorithm parameter
lk	link
lk_{C_i}	cellular link
$lk_{D_j} = d_j$	link d_j
l^{min}	minimun distance metric
$l_{d_{j_1},d_{j_2}}^{min}$	minimun distance metric between D2D d_{j_1} and D2D d_{j_2}
$\hat{l}_{d_{j_1},d_{j_2}}^{min}$	modified minimun distance metric between D2D d_{j_1} and D2D d_{j_2}
l_{u_i,d_j}^{min}	minimun distance metric between CUE u_i and D2D d_j
$\hat{l}_{u_i,d_j}^{min}$	modified minimun distance metric between CUE u_i and D2D d_j
L_q	average queue length
L_{u_i,b_k}	path-loss model from u_i to eNB b_k
M	feedback parameter (nr. of D2D pairs with the largest MDR metric)
m	blocklength
$M(b,c)$	CPP - matrix bxc that denotes the assignment of eNBs and SDN controllers
$M(b_k,c_j)$	CPP - denotes whether eNB b_k is assigned to SDN controller c_j

m_c	number of controller cores
$M(c,\theta)$	CPP - matrix $cx\theta$ that denotes the SDN controller location
$M(c_j,\theta_i)$	CPP - denotes whether SDN controller c_j is located in network entity θ_i
N	feedback parameter (nr. of subchannels)
n	cardinality of set N_{RB}
N_0	single-sided noise power spectral density
n'	RRA-URLLC-ACS-EM algorithm parameter - vertex
n_0	CPP-ABC algorithm parameter
N_i^k	ACS-EM algorithm parameter - neighborhood reachable by the ant
NL	ABC algorithm parameter - the operator neighbor list
NL_{size}	ABC algorithm parameter - length of the operator neighbor list
n_q	the request number received by the SDN controller
N'_{RB}	SA-RRA-ABC algorithm parameter
$N_{RB} = \{rb_k\} = \{rb_1, rb_2, \ldots, rb_n\}$	set of RBs
N_u	number of free RBs
$P = \{p_j\} = [p_1, p_2, \ldots, p_p]$	PSO algorithm parameter - position vector
p	PSO algorithm parameter - size of vector P
$pbest_j^i$	PSO algorithm parameter - the i particle's best position
P_{d_j}	transmitted power of D2D-capable User Equipment (DUEs)
$p(e)$	PSO algorithm parameter - probability of element e in A_p
$P^i = \{p_j^i\} = \left(p_1^i, p_2^i, \cdots, p_p^i\right)$	PSO algorithm parameter - position vector of particle i
p_i	ABC algorithm parameter - probabilistic value for solution i
p_K	the probability of K requests in the system
p_0	probability of no requests in the controller
P_T	the total power consumption
P_{u_i}	the transmit power of cellular user u_i
p_z	ACS-EM algorithm parameter - probability distribution
$Q^{-1}(\cdot)$	inverse of *Gaussian Q-function*
q	the number of bits sent by each D2D pair receiver to eNB
q_0	ACS-EM algorithm parameter
r_0	ACS-EM algorithm parameter
rnd_q, rnd_r, rnd_z	ACS-EM algorithm parameter - a uniformly distributed random variables
rnd_1, rnd_2	PSO algorithm parameter - a uniformly distributed random variables
R_D	ACS-EM algorithm rule (diversification)
r_{d_j}	the maximum number of bits transmitted on D2D link d_j
$r_{d_j}^{CM}$	the maximum number of bits transmitted on D2D link d_j in Cellular Mode (CM)

$r_{d_j^R}^{CM}$	the maximum number of bits transmitted on d_j^R in CM
$r_{d_j^T}^{CM}$	the maximum number of bits transmitted on d_j^T in CM
$r_{d_j}^{DM}$	the maximum number of bits transmitted on D2D link d_j in Dedicated Mode (DM)
$r_{d_j}^{RM}$	the maximum number of bits transmitted on D2D link d_j in Reuse Mode (RM)
$R_{d_j}^{w}$	the weighted throughput of D2D pair d_j
R_I	ACS-EM algorithm rule (intensification)
$R_{rb_n}^{RM}$	the channel rate of resource block rb_n in RM
R_T	system throughput
R_T^w	weighted system throughput
R_{u_i}	the throughput of the cellular link between CUE u_i and the respective eNB
r_{u_i}	the maximum number of bits transmitted on cellular link
$S = \{s_i\} = \{s_1, s_2, \ldots, s_s\}$	set of SGWs
s	cardinality of set S
SINR_{d_j}	received SINR of d_j^R
$\text{SINR}_{d_j}^{RM}$	received SINR of d_j^R in RM
SINR_d^t	SINR threshold of D2D receiver
SINR_{u_i}	received SINR of eNB when CUE u_i transmits
SINR_u^t	SINR threshold when a CUE transmits
SN	ABC algorithm parameter - food sources
SNR_{d_j}	received SNR of d_j^R
$\text{SNR}_{d_j}^{CM}$	received SNR of d_j^R in CM
$\text{SNR}_{d_j^R}^{CM}$	received SNR of eNB when d_j^R transmits in CM
$\text{SNR}_{d_j^T}^{CM}$	received SNR of eNB when d_j^T transmits in CM
$\text{SNR}_{d_j}^{DM}$	received SNR of d_j^R in DM
SNR_{u_i}	received SNR of eNB when CUE u_i transmits
$t_{b_k c_j}$	CPP - propagation latency between the eNB b_k and the SDN controller c_j
$t_{b_k \theta_i}$	propagation latency between the eNB b_k and the controller located in θ_i
t_{c_j}	CPP - the average response time of the SDN controller c_j
t_{QoS}	CPP - Quality of Service (QoS) time
T_r	CPP - total average request response time (considering all SDN controllers
$t_{r_{c_j}}$	CPP - the average request response time of a SDN controller c_j
$T_r(t_{r_{c_j}})$	CPP - cost function associated with the average request response time of the SDN controller c_j
$U = \{u_i\} = \{u_1, u_2, \ldots, u_u\}$	set of CUEs

u	cardinality of set U
V	channel dispersion
V_{b_k,c_j}	CPP-ACS-EM algorithm parameter - vertex subset
V_{c_j,θ_i}	CPP-ACS-EM algorithm parameter - vertex subset
V^d	RRA-URLLC-ACS-EM algorithm parameter - vertex subset
$v^u_{i,k}$	RRA-URLLC-ACS-EM algorithm parameter - vertex
$v^d_{j,k}$	RRA-URLLC-ACS-EM algorithm parameter - vertex
$v^d_{j,n'}$	RRA-URLLC-ACS-EM algorithm parameter - vertex
V_{tx}	set of vertices
V^u	RRA-URLLC-ACS-EM algorithm parameter - vertex subset
W_{RB}	system bandwidth per RB
W	system bandwidth
w_{min}, w_{max}	PSO algorithm parameters
WNL	ABC algorithm parameter - the winning neighboring list
X	feedback parameter
$\hat{X}$	feedback parameter
x_{d_j}	indicator denoting the communication mode used by D2D links
Y	feedback parameter
$\hat{Y}$	feedback parameter
Z	ACS-EM algorithm parameters
z_0	ACS-EM algorithm parameters

Greek Symbols

α	ACS-EM algorithm parameter - establishes the relative importance of pheromone tracks
$\alpha_1, \alpha_2, \alpha_3, \alpha_4$	RRA - weights of objective function
β	ACS-EM algorithm parameter - relative importance of heuristic information
$\beta_{d^T_j,d^R_j}$	the fast fading gain from d^T_j to d^R_j
$\beta_{d^T_{j_2},d^R_{j_1}}$	the fast fading gain from $d^T_{j_2}$ to $d^R_{j_1}$
$\beta_{d^T_{j_1},d^R_{j_2}}$	the fast fading gain from $d^T_{j_1}$ to $d^R_{j_2}$
β_{u_i,b_k}	the fast fading gain from u_i to eNB b_k
β_{u_i,d^R_j}	the fast fading gain from u_i to eNB d^R_j
γ_{d_j}	q-bit feedback
$\gamma_c, \gamma_T, \gamma_\pi, \gamma_F$	CPP - weights of each cost function in objective function
δ	ACS-EM parameters - pheromone evaporation coefficient
Δu_i	the set of D2D pairs that reuse the RBs allocated for u_i
$\Delta\tau$	ACS-EM parameters - increment in the pheromone
ϵ	decoding error probability

$\zeta_{d_j^T,d_j^R}$	the slow fading gain from d_j^T to d_j^R
$\zeta_{d_{j_2}^T,d_{j_1}^R}$	the slow fading gain from $d_{j_2}^T$ to $d_{j_1}^R$
$\zeta_{d_{j_1}^T,d_{j_2}^R}$	the slow fading gain from $d_{j_1}^T$ to $d_{j_2}^R$
ζ_{u_i,b_k}	the slow fading gain from u_i to eNB b_k
ζ_{u_i,d_j^R}	the slow fading gain from u_i to d_j^R
$\Theta = \{\theta_i\} = \{\theta_1, \theta_2, \ldots, \theta_\theta\}$	CPP - set of possible locations of the controllers
θ	CPP - cardinality of set Θ
Λ	set of arcs
λ	the average control requests rate generated by each D2D pair
λ_c	the average control requests rate received in the SDN controller c_i
λ_{eff}	effective request rate in the SDN controller
λ_{lost}	lost requests rate in the SDN controller
$\lambda_{lost}^{threshold}$	the threshold of lost requests rate
μ	processing rate of each core
ξ	CPP - constant factors that specify the increase/decrease rate of the cost function $T_r(t_{r_{c_j}})$
Π	CPP - the input to the problem that defines the search space
π_C^{avglat}	CPP - cost function associated with the controller location
ρ	controller utilization rate
σ^2	noise variance
τ	ACS-EM algorithm parameter - amount of pheromone
τ_{ij}	ACS-EM algorithm parameter - amount of pheromone between nodes v_{txi} and v_{txj}
τ_0	ACS-EM algorithm parameter - initial pheromone value
$\Upsilon^i = \{\upsilon_j^i\} = \left(\upsilon_1^i, \upsilon_2^i, \cdots, \upsilon_p^i\right)$	PSO algorithm parameter - velocity vector of particle i
Φ	CPP - set of constraints
$\psi[u_i, d_j]$	indicator expressing the reuse of spectral resources
$\Omega = \{1, 2, \ldots, J\}$	ACS-EM algorithm parameter - set of nodes that belong to the path traveled by the *best-so-far* ant
ω	PSO algorithm parameter - inertia
ω_{d_j}	the strength of social relationships for a pair of users d_j^T and d_j^R
$\omega_{d_j^R}^1$	features set of d_j^R
$\omega_{d_j^T}^1$	features set of d_j^T
$\omega_{d_j}^1$	common features index
$\omega_{d_j^R}^2$	friends set of d_j^R
$\omega_{d_j^T}^2$	friends set of d_j^T
$\omega_{d_j}^2$	common friends index
$\omega_{d_j^R \text{->} d_j^T}^r$	indicator expressing if user d_j^R follows user d_j^T (reciprocity)

$\omega^r_{d_j^T \text{->} d_j^R}$	indicator expressing if user d_j^R follows user d_j^T (reciprocity)
$\omega^3_{d_j}$	reciprocity index

ACRONYMS LIST

Acronyms

5G	5th generation
ABC	Artificial Bee Colony
AC	Ant Colony
ACA	adaptive controller assignment
ACO	Ant Colony Optimization
ACS	Ant Colony System
ACS-EM	Ant Colony System with External Memory
AFS	Artificial Fish Swarm
ANN	Artificial Neural Network
API	Application Programming Interface
AWGN	Additive White Gaussian Noise
BalCon	Balanced Controllers
BFO	Bacterial Foraging Optimization
BFS	brute-force search
BGP	Border Gateway Protocol
BOA	Butterfy Optimization Algorithm
BS	Base Station
CAP	Controller Assignment or Allocation Problem
CCLGA	cardinality-constrained locally greedy algorithm
CCSAA	cardinality-constrained subchannel assignment algorithm
CG	Conflict Graph
CLP	Controller Location Problem
CM	Cellular Mode
CPA	Controller Provisioning Algorithm
CPP	Controller Placement Problem
CPP	Controller Placement
CPP-ABC	CPP Artificial Bee Colony
CPP-ACS	CPP Ant Colony System
CPP-ACS-EM	CPP Ant Colony System with External Memory
CPP-PSO	Particle Swarm Optimization
CPU	Central Process Unit
CSC	Core SDN Controller
CSI	Channel State Information
CTR	Controller
CUE	Cellular User Equipment

D2D	Device-to-Device
DCA	Dynamic Controller Assignment or Allocation
DCAP	Dynamic Controller Assignment or Allocation Problem
DL	Downlink
DM	Dedicated Mode
DUE	D2D-capable User Equipment
EE	Energy Efficiency
eMBB	Enhanced Mobile Broadband
eNB	evolved NodeB
ES	Exhaustive Search
FA	Firefly Algorithm
FIFO	First In First Out
GBA	Graph-Based Allocation Algorithm
GC	Global Controller
GRAA	Greedy Resource Allocation Algorithm
GTM+	Greedy Throughput Maximization Plus
GWO	Grey Wolf Optimizer
HSS	Home Subscriber Server
IIoT	Industrial Internet of Things
INLP	Integer Non-Linear Programming
IoT	Internet of Things
ITS	Intelligent Traffic Systems
LC	Local Controller
LCMA	Low Complexity Master Assignment
LGA	locally greedy algorithm
LSC	Local SDN Controller
LTE	Long Term Evolution
LTE-A	Long Term Evolution Advanced
MDR	Maximum Distance Ratio
MEC	Mobile Edge Computing
MHNSGA	Multi-start hybrid non-dominated sorting genetic algorithm
MILP	Mixed Integer Linear Programming
MINLP	Mixed Integer Non-Linear Programming
MIP	Mixed Integer Programming
MISO	multiple-input-single-output
ML	Machine Leaning
MME	Mobile Management Entity
mMTC	Massive Machine Type Communication
MOGA	Multi-Objective Genetic Algorithm
MSE	Mean Squared Error
NEE	Network Energy Efficiency

NFL	no-free-lunch
NFV	Network Function Virtualization
NLP	Nonlinear Programming Problem
NN	Neural Network
NOMA	on-orthogonal multiple access
NP	nondeterministic polynomial
OF	Objective Function
OFDMA	orthogonal frequency-division multiple access
OMA	orthogonal multiple acces
PCRF	Policy and Charging Rules Function
PDS	deep post-decision state
PGW	Packet Data Network Gateway
POCO	Pareto-based Optimal COntroller-placement
ProSe	Proximity Services
PSO	Particle Swarm Optimization
PSOGSA	Particle Swarm Optimization Gravitational Search Algorithm
QoS	Quality of Service
RASCAR	recovery-aware switch-controller assignment and routing
RB	resource block
RF	radio frequency
RL	Reinforcement Learning
RM	Reuse Mode
RQs	research questions
RRA	Radio Resource Allocation
RRA-ACS	Radio Resource Allocation based on Ant Colony System
RRA-URLLC-ABC	Artificial Bee Colony
RRA-URLLC-ACS-EM	Ant Colony System with External Memory
RRA-URLLC-Greedy	Greedy Resource Allocation Algorithm
RRA-URLLC-PSO	Particle Swarm Optimization
RRM	Radio Resource Management
SA-GRAA	Social-aware Greedy Resource Allocation Algorithm
SA-RRA-ABC	Social-aware RRA Artificial Bee Colony
SAA	Switch Assignment Algorithm
SDN	Software Defined Networking
SGW	Serving Gateway
SINR	Signal-to-Interference-plus-Noise Ratio
SM	shared mode or underlay mode
SNR	Signal-to-Noise Ratio
STP	Successful Transmission Probability
SW	Switch
TDD	time division duplex

TLBO	Teacher Learning Based Optimization
UE	User Equipment
UL	Uplink
URLLC	Ultra-reliable and Low Latency Communications
V2V	vehicular to vehicular
VBO	Varna Based Optimization
VM	virtual machine
WOA	Whale Optimization Algorithm
XMPP	Extensible Messaging, and Presence Protocol

1 INTRODUCTION

The rapid development of smart devices has led to an exponential growth in the demands for services with high data consumption. Applications with high data transmission rate are becoming increasingly prevalent, imposing a new traffic pressure on the cellular network. Furthermore, with the advent of the Internet of Things (IoT), billions of devices will be connected to the Internet, producing data to be transported. The next generation of mobile technologies is expected to connect people, things, data, applications, transport systems, and cities in smart net worked communication environments, as well as transport a considerable amount of data much faster, reliably connect a huge number of devices, and process very high volumes of data with minimal delay [1].

5th generation (5G) communications have been designed to offer both more advanced and complex configurations aimed at better performance towards meeting the requirements of society and a new form of thinking about the components that comprise them. Such technology will provide unlimited access to information, as well as availability for its sharing anywhere, at any time, by any person or thing, for the benefit of society [2].

Next generation wireless networks are characterized by numerous devices, interconnected networks, and by an increase in the data traffic in comparison to the current one. Other characteristics include seamless integration of heterogeneous systems, use of femtocells, and increased capacity and performance with reduced latency [3, 4]. Mobile networks, ultra-dense networks, Device-to-Device (D2D) communications, ultra-reliable communications, and mass communication of machines have also been considered.

D2D communications are defined as direct communications between two mobile devices that do not pass through a cellular Base Station (BS) or network core [5]. They can have important applications in 5G [6, 7], extending the coverage of mobile networks and offering multiple services based on the proximity of the devices (e.g., local information exchange, multimedia traffic in IoT systems, gaming, social applications, vehicle-to-vehicle communications in Intelligent Traffic Systems (ITS), among others). Recent research [6] focused on the benefits of D2D communications for meeting the requirements imposed by 5G networks, such as higher data rate, low delay and energy consumption, coverage extension, reliability in communications (even in case of network failure), traffic offloading, and possibility for instantaneous communications between devices.

Such communications require different levels of control by the operator. Based on the business model, they have either full or partial control over the resource allocation among source, destination, and relay devices, or no control. Below are the different types of D2D communications defined according to the function of the cellular infrastructure [8]:

1. D2D communications with the establishment of an operator-controlled connection: source

and destination devices talk and exchange data with no BS; however, they are served by a BS to establish the link.

2. D2D communications with the establishment of a device-controlled connection: source and destination devices communicate directly with each other with no operator control and must implement methods that guarantee limited interference with other devices on both the same layer and macrocell layer.

The D2D links use either licensed or unlicensed spectrum, in inband and outband communications. In inband D2D communication, both D2D and cellular communication links use licensed spectrum. In this type of communication, a challenge is to handle D2D and cellular interference. In outband D2D communication, D2D communication uses unlicensed spectrum, whereas cellular uses its own dedicated licensed spectrum. In this type of communication, no interference issue between cellular and D2D is considered [9, 10].

D2D communications with establishing/managing connection through the cellular infrastructure were considered in this study. Among the advantages of such an approach are possible data offloading between the mobile network and the D2D network and an efficient spectrum allocation of the spectrum if inband D2D is used.

Other important design principles for a future wireless access is the decoupling of user data and a functional control system. The latter includes information and necessary procedures for the management of devices' access to the network. Therefore, the Software Defined Networking (SDN) paradigm can provide a better mechanism for network control, improving management efficiency.

SDNs approach the creation of networks whose control is detached from the hardware and given to a software application called SDN controller. An SDN network architecture promotes a more programmable, automatable, and flexible separation of the control and data planes, and the network is virtualized and independent of the underlying physical infrastructure.

The main advantages of SDN include: (i) an agile and flexible network that can adapt automatically through the use of standard software interfaces, (ii) an easy implementation of complex network functions through algorithms defined by users and implemented in software in the controller, (iii) a rapid deployment of new network features, and (iv) manufacturer's independence in network devices. Since the intelligence of the network is centralized in the controller, the network devices, which are only handlers of the data flows, can be obtained from multiple manufacturers.

Our proposal employs one or more SDN controllers in the cellular infrastructure for the management of ongoing D2D communications in the coverage area of each evolved NodeB (eNB). Each controller maintains registrations and information on the location of users and their possible establishment of D2D communication. The information enables the controller to determine the beginning of D2D communications and send messages to the devices involved for beginning/establishing a direct connection between them and allocating network resources.

The controller also monitors the state of the D2D communication. If the link is lost (for example, due to a change in location), it reconfigures the User Equipment (UE) and the network entities to return the communication to the cell infrastructure and guarantees the continuity of the sessions. Therefore, it can implement efficient algorithms of data offloading and radio resource allocation on the application layer and communicate with UE, eNB, and Serving Gateway (SGW) to define the path of the data packets by protocols, such as OpenFlow. The proposed solution can provide support for many new application scenarios, such as smart city, Internet of Things (IoT), Industrial Internet of Things (IIoT) applications, factory automation and control for Factories of the Future, autonomous vehicles, smart grid automation, unmanned aerial vehicles control information delivery, local information exchange, multimedia traffic in IoT systems, gaming, social applications, vehicle-to-vehicle communications in Intelligent Traffic Systems (ITS), among others.

Radio Resource Allocation (RRA) in cellular systems is a relevant and difficult task that must allocate the system resources in the most efficient way while fulfilling different constraints such as Quality of Service (QoS) [11, 12]. The optimal allocation of radio resources is one of the fundamental challenges for the design and operation of cellular wireless networks. Resource allocation problems have to be solved for specific network scenarios, taking into account rapidly changing wireless channels and QoS requirements for users. This doctoral thesis proposes two solution methods for the RRA problem. The first deals with the intensity of social relations between users in a socially aware radio resource allocation problem, determining the number of D2D communications allowed in the cellular network. The second addresses the resource allocation problem in a D2D cellular network utilizing Ultra-reliable and Low Latency Communications (URLLC) services to send short packets directly to their destination.

The widespread dissemination of social networks (such as Facebook, What'sapp, Instagram, Twitter, among others) has required the adequate collection and processing of the demands by each mobile telecom operator. The possibility of using control information on the usage of such social network information enables devices of some computational capacity to identify interesting or related peers for the exchange of information and context-based services, among other options. The wide adoption of social networks has given rise to a new dimension in the design of cellular communication systems, and the exploration of social interaction patterns is expected to significantly improve D2D peer discovery, resource allocation, data offloading, and security [13]. Therefore, this study considers the intensity of social relations among users in a social-aware radio resource allocation solution that determines the number of D2D communications admitted to the cellular network.

On the other hand, systems with growing demands (e.g., mobile networks) must be designed with high scalability and reliability for avoiding problems (e.g., single point of failure). Given the huge quantity of traffic to be treated by 5G networks and since a single centralized SDN controller can lead to architectures with such problems, the following research questions (RQs) have emerged:

1. How many controllers are required to manage all D2D communications?
2. How can eNBs be assigned to the controllers?
3. What is the best location of SDN controllers within the cellular network?

The minimum number of SDN controllers required, their optimal location, and the optimal assignment of eNBs to them must be determined so that the RQs can be answered. Therefore, this study proposes a method that considers two stages, according to which the number of D2D communications that maximizes throughput in the network is obtained and, based on such information, the Controller Placement Problem (CPP) is solved.

In the first stage, a social-aware radio resource allocation solution proposed obtains the number of D2D communications that maximizes throughput on the network, since the interference limits are respected in both cellular and D2D communications. The solution was compared with a social-unaware solution.

In the second stage, the CPP is modeled as an optimization problem that considers the minimization of the weighted sum of three metrics, namely number of SDN controllers, response time, and eNB-controller assignment that enables an efficient management of all D2D communications in a Long Term Evolution Advanced (LTE-A) network. The Ant Colony System (ACS) and Ant Colony System with External Memory (ACS-EM) meta-heuristics [14, 15] were used in the CPP solution.

Moreover, as aforementioned, the radio resource allocation problem is studied with the modeling of two optimization problems. In the first, it is formulated for maximizing the overall system throughput by enabling more D2D pairs to reuse a same resource block (RB), subject to the constraint of the maximum tolerable interference level at each RB. In the second, the optimization problem aims at maximizing system energy efficiency through an efficient use of spectrum while maintaining QoS requirements in cellular and D2D communications.

Firstly, the RRA approach investigates the resource allocation problem in which more than one D2D pair can share a same RB with Cellular User Equipment (CUE). RBs are assigned to CUEs and the QoS of both CUEs and D2D-capable User Equipment (DUEs) is considered. Mutual interferences among different D2D pairs, as well as that between DUE and CUE sharing the same RB are taken into account. Moreover, the achievement of high spectral efficiency while ensuring no excessive interference is caused to CUEs depends on an adequate interference-aware allocation of RBs to D2D users. In this scenario, the eNBs must have full or partial Channel State Information (CSI) to perform an efficient resource allocation.

In the full CSI model, the eNB knows the perfect CSI of all links [16, 17, 18], whereas in practice, it knows only the channel power gains of CUEs for eNB and D2D transmitter (d_j^T) for eNB links - a priori, it does not know the channel power gains of CUE for D2D receiver (d_j^R) and d_j^T for d_j^R links. Therefore, the CSI of such links can be obtained through feedback, which causes high overhead [17, 18], especially when the number of CUEs and D2D pairs is large.

In the feedback model, the D2D receiver sends information to the eNB about the interference channel gain from the CUE to the d_j^R and the channel gain between D2D transmitter (d_j^T) and D2D receiver (d_j^R). Considering *N* cellular terminals and *M* D2D pairs, a scheme named selected-*NM* Maximum Distance Ratio (MDR) q-bit feedback scheme is proposed in the approach based on partial CSI. The scheme considers the need for reducing the network overhead generated by feedback and limits the sending of q-bits of channel information and the number of links. Consequently, it ensures both *N* cellular and *M* D2D pair links can be possible partners to share radio resources, guaranteeing QoS parameters based on the Maximum Distance Ratio metrics [17]. The effect of feedback information on the performance of D2D underlaying cellular networks is studied considering multiple D2D pairs for RB.

The second RRA deals with the resource allocation problem in a D2D cellular network that uses URLLC services to send short packets directly to their destination. Since such packets are essential for low latency critical mission communications, the channel coding rate is adopted as a performance metric in D2D links in the finite packet length regime, as defined in [19]. It provides the maximum number of data bits sent in a packet of size m and has been used as a performance metric in systems with URLLC services [20, 21, 22, 23, 24, 25].

In URLLC communications, the decoding error probability should be lower than 10^{-5} and no higher than 1 ms latency for a small packet size (e.g., 32 bytes) [26]. Ultra-high reliability and low latency are the major technical challenges in URLLC networks [20, 21, 22, 25, 24, 23], with a tradeoff between them. Specifically, reliability can be improved by techniques such as signaling and relaying, which increased latency [20]. On the other hand, some wireless networks focus on long packet transmission scenarios to maximize system throughput and favor energy efficiency. The simultaneous assurance of high reliability and low latency is challenging, which highlights the need for new technologies that meet URLLC service requirements.

As a traditional metric in information theory, Shannon capacity has been widely used to characterize the maximum achievable transmission rate [20, 21, 22], considering channels with Additive White Gaussian Noise (AWGN). Channel capacity bounds determine the maximum data rates to be transmitted with low error probability, assuming no constraints on delay and encoder and decoder complexity. However, in scenarios which require the use of short packets, the impact of packet size and decoding error is more prominent and cannot be ignored in the aforementioned channel capacity metric [25, 27].

This approach assigns RBs to UEs and considers the QoS of both CUE and DUE in a scenario where CUEs use conventional network services and DUEs use URLLC ones. The formulation involves the following constraints: two or more D2D pairs can share a same RB with cellular user; an upper limit on the tolerable interference level must be considered in each RB; and a minimum Signal-to-Interference-plus-Noise Ratio (SINR) must be respected for the cellular user and the D2D pairs. Considering such constraints and two adequately defined models (system model and interference model), the problem establishes the maximization of the network energy efficiency (η_{EE}) as the objective.

1.1 MOTIVATION

Data consumption has currently faced a growing demand in cellular networks. The dissemination of applications that require better quality of service has led to the evolution of the current cellular networks to the next generation (5G). Some of the promises of the 5G networks include faster data throughput, higher bandwidth, lower latency, and the assimilation and adoption of other technologies. Such a scenario, therefore, has required studies of new technologies to that the goals pursued in 5G become a reality.

One of such technologies is D2D communications, which enables nearby UEs to communicate without involving an eNB in data traffic. Using a frequency band licensed for mobile devices, two nearby users can exchange messages directly, without passing through a BS. However, the consideration of such technology has imposed new challenges such as:

- interference control: UEs involved in a D2D communications may suffer, and generate, interference to other UEs present in a same cell;
- delay-sensitive processing: the transference of audio, video, or online games requires real-time processing and is delay-sensitive. Consequently, such requirements must be considered in the search for treatments of real or near-real time traffic;
- resource allocation: D2D communications requires a certain amount of resources, such as bandwidth and channels. However, the necessary ones must be allocated so that other cells do not suffer interference, maintaining efficient communication.
- communication management: the availability of D2D communications supervision/management promotes, for example, an efficient search for a communicating pair, identification of connection loss issues, efficient interference control, and allocation of resources.

Other challenges in 5G network deployment are scalability and flexibility. Future cellular network infrastructures and methodologies should enable the integration of heterogeneous networks in a big network ecosystem. Besides, a large number of users can request several services at the same time, meaning the 5G network must support a scalable demand. Therefore, the use of technologies such as Network Function Virtualization (NFV) and SDN has been proposed for solving those problems.

Software-defined networks have an agile network architecture that separates management from the underlying network infrastructure and enables administrators to dynamically adjust the flow of traffic across the network towards meeting changing needs. Moreover, it reduces the complexity of statistically defined networks, automates network functions, accelerates the implementation of applications and services, and simplifies the implementation and administration of network resources.

Network scalability requires scenarios where multiple SDN controllers manage communications and control functions in LTE-A networks, thus raising new questions, such as:

- how many controllers are required to manage all D2D communications?;
- how can eNBs be assigned to the controllers?; and
- what is the best location of SDN controllers within the cellular network?.

Such challenges have motivated our proposal of a model that solves the controller placement location issue in a scenario that considers D2D communications, the radio resource allocation, as well as the interference between cellular and D2D communications. According to our literature review on the SDN CPP problem for wireless networks, the treatment of D2D communications has not been addressed, and the Artificial Bee Colony (ABC) and ACS-EM meta-heuristics have not been applied to the problem considered here.

On the other hand, modern society is moving towards total connectivity and people's relationships in the real world has been transferred to the interconnection and exchange of information in the virtual world. Social networking services have facilitated the active communication of billions of people and their wide adoption has broadened connections among people. In this context, technological advancements have led researchers to find out the interaction patterns among users, and the exploitation of these patterns can contribute to significantly improve the radio resource allocation for D2D communications.

Social-aware D2D communications refer to the finding of interaction patterns among nearby social network users and exploitation of such patterns for the design of efficient communication and radio resource allocation in D2D networks. Therefore, studies of scenarios that also consider social relationships among users communicating through a direct link (D2D) and managed by SDN controllers in an LTE-A network have drawn the interest of researchers.

In this sense, the intensity of user's social relationship must be considered for the determination of number of the D2D communications in an LTE-A network jointly with other aspects, such as cellular spectrum reuse and interference thresholds. Therefore, an efficient radio resources allocation is required so that they can be shared between D2D and cellular communications.

Due to the quantity of D2D communications, a model that obtains the minimum number of controllers and their physical location while performing a balanced assignment of eNBs to SDN controllers is required.

The CSI available in eNB and D2D pairs must be considered when radio resource allocation is addressed. Since the eNB does not participate in the sending of D2D communication data, it has no information about the gain or interference of the link between transmitter and D2D receiver. Therefore, the receiver of the D2D communication must send such information to eNB, which generates feedback overhead. Efficient feedback methods must be implemented for decreasing overhead. Accordingly, we have proposed a limited feedback model with partial CSI that uses the selected-NM method. Instead of the D2D receiver sending information from all CUEs and D2D pairs, it sends it from a specific number of cellular users and D2D pairs that are most promising for the shared RB.

On the other hand, 5G cellular networks must support a number of emerging applications with varying QoS requirements, leading to services categorized as Enhanced Mobile Broadband (eMBB), Massive Machine Type Communication (mMTC), and URLLC [28]. 5G wireless technologies are expected to improve performance towards mission-critical applications (e.g., factory automation and control for Factories of the Future, autonomous vehicles, smart grid automation, unmanned aerial vehicles control information delivery, and remote surgery [29, 30]). Such applications have strict latency and reliability requirements, which can be met by URLLC services, since they support real-time, ultra-reliable, and low latency communications [29, 28, 31].

The simultaneous assurance of high reliability and low latency is challenging, thus highlighting the need for new technologies that meet URLLC service requirements. The integration of D2D into future industrial wireless networks and next-generation manufacturing can support massive machine-type connections. A D2D communication system can satisfy URLLC service requirements, as well as improve spectrum efficiency, energy efficiency, and overall network capacity, which has motivated our study of the radio resource allocation with the use of URLLC services by D2D communications.

1.2 OBJECTIVES

1.2.1 General Objective

The general objective of this research is to solve the Controller Placement (CPP) and the Radio Resource Allocation (RRA) problems in a cellular network that requires D2D communications management.

1.2.2 Specific Objectives

The specific objectives involve:

- The study and understanding of the basic concepts of D2D, radio resource allocation, interference modeling, and metaheuristics.
- The study and understanding of the basic concepts of social relationships and their influence on the resource allocation in wireless communication networks.
- The study and understanding of the basic concepts of CSI and feedback scheme and their influence on the resource allocation in wireless communication networks.
- The study and understanding of the basic concepts of URLLC services and their influence on the resource allocation in wireless communication networks.
- The learning and evaluation of different Controller Placement Problem (CPP) solution proposals.

- The learning and evaluation of different RRA solution proposals.
- The understanding and evaluation of different proposals that use social networks in radio resource allocation considering the D2D underlay model.
- The understanding and evaluation of different proposals that use URLLC services in radio resource allocation considering the D2D underlay model.
- The proposal and evaluation of a model that enables studies of the influence of social-aware D2D communications management on the number of SDN controllers, assignment of eNB to controllers, and their location when interference and radio resource allocation are considered.
- The proposal and evaluation of a model that enables studies of social-aware radio resource allocation in a scenario of D2D communications.
- The selection, application and evaluation of bioinspired meta-heuristics for solving the controller placement problem.
- The proposal and evaluation of a model that enables studies of the influence of limited feedback with partial CSI on the radio resource allocation considering D2D communications.
- The proposal and evaluation of a model that enables studies of the influence of URLLC services on the radio resource allocation considering D2D communications.
- The selection, application, and evaluation of bioinspired meta-heuristics for solving the radio resource allocation problem.

1.3 CONTRIBUTIONS

The following contributions must be highlighted:

- management of social-aware D2D communications by multiple controllers;
- solution that determines the number of D2D communications considering the social relationship among users and the interference between cellular and D2D communications;
- an integrated approach that solves the CPP problem combining D2D communications and controllers management and considering the social-aware radio resource allocation;
- use and evaluation of the ABC metaheuristic for solving the controller placement problem;
- use and evaluation of the ant colony system with external memory (ACS-EM) metaheuristic for solving the controller placement problem;
- a statistical evaluation of ABC and ACS-EM meta-heuristics;

- modeling and use of the response time of a controller during D2D communications management in LTE-A cellular networks;
- design of an approach to RRA that integrates physical and social layers, partial CSI, and a limited feedback scheme towards maximizing system throughput;
- proposal of a feedback scheme in which a D2D pair feeds back to eNB its SINR on N subchannels, considering the interference of M D2D pairs with the largest MDR metric;
- design of an approach to RRA that integrates URLLC services, power control, and communication mode selection towards maximizing the network energy efficiency;
- modeling and use of metrics that evaluate the impact of URLLC services on network energy efficiency, system throughput, and transmission power; and
- use and evaluation of the ABC metaheuristic in solving radio resource allocation.
- use and evaluation of the ACS-EM metaheuristic in solving radio resource allocation.
- use and evaluation of the Particle Swarm Optimization (PSO) metaheuristic in solving radio resource allocation.

1.4 THESIS STATEMENT

Cellular networks are evolving to the 5th generation, which requires the use of new technologies that offer users higher data transfer rates and continuous connection. In this scenario, technologies such as D2D communications will be widely used. The proposal of a method that helps the determination of number of controllers and their location can contribute to the design of an adequate management model in the next-generation cellular networks. New methods that promote a better use of radio resources must also be studied, since they represent a limited and widely demanded resource.

This thesis proposes methods that help an efficient design of the control plan in cellular networks. An optimization problem that leads to the smallest number of controllers that can efficiently manage the network, as well as their best location in the cellular infrastructure, ensuring QoS was modeled.

This study also analyzes methods towards improving the use of radio resources when D2D communications are supported on cellular networks. Given a previous cellular-based resource allocation, an allocation method is proposed for the reuse of resources by the greatest possible number of D2D communications, without affecting cellular communications.

1.5 METHODOLOGY

This section describes the methodology for the solution of CPP and RRA problems, according to the steps followed for each problem.

1.5.1 Controller Placement Problem

The CPP for the management of D2D communications in an LTE-A network was approached according to the following methodology.

Definition of the scenarios and network models of interest:

Scenario of D2D communications with licensed bandwidth sharing in underlay mode and SDN-based network architecture subject to a high demand, requiring the use of multiple controllers. The modeling of an optimization problem is introduced in order to minimize an objective function that takes into account the weighted sum of the number of controllers, the response time, the average propagation latency between the controllers and the assigned eNBs, and a fairness parameter between the controllers' utilization. A set of mathematical equations representing the strength of social relationships, the channel model, the interference model, and the queueing model for D2D controllers was developed.

Formal definition of the controller placement problem:

A set of mathematical equations representing the behavior of SDN LTE-A networks was developed for the definition of the CPP. Different constraints such as response time, lost request rate, eNB-controller assignment, and controller location were considered due to their importance for the design of LTE-A networks with multiple SDN controllers. The model aims at minimizing the weighted sum of number of controllers, response time, average propagation latency between the controllers and the assigned eNBs, and justice parameter while determining both number and location of controllers and the eNB-controller assignment.

Obtaining of number of D2D communications:

A set of mathematical equations representing interference in each eNB when CUEs and DUEs use the same radio features was developed for calculating the number of D2D communications allowed in the network. Different constraints such as SINR and D2D reuse resource block quantity were considered due to their importance when the RRA is performed in underlay mode. The goal is to obtain the number of supported D2D communications when only interference between links is considered and social relations between users are taken into account. Two greedy heuristics that returned the number of D2D communications in the cellular network solved the problem.

Solution of the controller placement problem:

The solution considered number of controllers, their location, and eNB-controller assignment. ABC and ACS-EM metaheuristics solved the model and they returned the near to optimal solution in acceptable time.

Study of the results of the controller placement problem:

Four bio-inspired meta-heuristics in two scenarios solved the CPP problem and the influence of the D2D request rate on number, location, and assignment of SDN was analyzed. The result analysis starts with a small number of control requests, which is increased for investigations on the way the model reacts. Variations in the number of controllers necessary for the management of communications in both social-unaware and social-aware scenarios were studied.

Study of the meta-heuristics' performances:

Statistical tests evaluated the performance of the proposed ABC and ACS-EM meta-heuristics when paired with the ACS and PSO meta-heuristics. 32 simulations were performed for each case and Kolmogorov-Smirnov, Kruskal-Wallis, and Wilconxon hypothesis tests determined the meta-heuristic that performed best in the CPP solution. Both computational complexity and number of computations for the four meta-heuristics were analyzed.

1.5.2 Radio Resource Allocation Problem

The RRA for D2D communications in a cellular network was approached according to the following methodology.

Definition of the scenarios and network models of interest:

A 1st scenario of D2D communications is treated in the scope of a single cell with channel sharing in underlay mode and partial CSI. Regarding a 2nd scenario, also in a single cell and based on D2D communications, URLLC services are offered and the communication mode is chosen among options such as cellular, reuse (or underlay), and dedicated (or overlay). The modeling of an optimization problem is introduced towards maximizing network throughput and evaluating the impact of social relationship between users and the partial CSI on radio resource allocation. A set of mathematical equations representing the strength of social relationships and the Maximum Distance Ratio (MDR) metric was developed. MDR was considered in the proposed limited feedback with partial CSI scheme and an ABC-based meta-heuristic solved the problem. The meta-heuristic returned the number of D2D communications in the cellular network and the channel allocation performed in reuse mode.

The modeling of an optimization problem is introduced towards maximizing energy efficiency in the network and evaluating the impact of URLLC services on radio resource allocation. First, a set of mathematical equations representing the blocklength and decoding error probability was developed. The optimization problem considers communication mode selection, power control, and channel allocation.

Formal definition of the radio resource allocation problem:

A set of mathematical equations representing the social network model, the channel model, the interference model, the CSI and Limited selected-NM Feedback model, and the URLLC services model was developed for defining RRA. Different constraints such as minimum SINR and maximum transmission power were considered due to their importance for channel allocation. Two optimization problems - one for maximizing network throughput and the other for maximizing network energy efficiency - were defined.

Solution of the radio resource allocation problem:

To solve the first RRA optimization problem, a approach based on ABC meta-heuristic was proposed . It returned the near to optimal solution in acceptable time solved the models. To solve the second RRA optimization problem, three meta-heuristics, namely Artificial Bee Colony (RRA-URLLC-ABC), Ant Colony System with External Memory (RRA-URLLC-ACS-EM), and Particle Swarm Optimization (RRA-URLLC-PSO) were proposed. They returned the communication mode, transmit power, and channel allocation for both cellular and D2D communications, as well as the number of D2D communications in the cellular network.

Study of the results of the radio resource allocation problem:

Bio-inspired meta-heuristics used in two scenarios solved the RRA problem. In the first, the influence of feedback parameters on the system throughput and the feedback overhead of the proposed scheme were analyzed and the solution was compared with two greedy ones considering full and partial CSI. In the second, the influence of the parameters associated with URLLC services on the throughput and the impact of varying the number of D2D pairs on the network energy efficiency were analyzed. The solution was compared with an exhaustive search algorithm for the validation of its effectiveness. The computational complexity of the meta-heuristics was also investigated.

1.6 ORGANIZATION

The remainder of this thesis is organized as follows: Chapter 2 presents the state-of-the-art of CPP and RRA solutions and their categorization and comparison; Chapter 3 focuses on the modeling of the system and describes ABC, ACS-EM and PSO meta-heuristics.

Chapter 4 proposes a solution to CPP, describing ABC and ACS-EM meta-heuristics for CPP. Chapter 5 addresses the proposed solutions to the RRA problem, describing Social-aware RRA

Artificial Bee Colony (SA-RRA-ABC), RRA-URLLC-ABC, RRA-URLLC-ACS-EM and RRA-URLLC-PSO meta-heuristics for RRA.

Chapter 6 reports on analyses of the results; finally, Chapter 7 concludes the research and suggests some future work.

2 RELATED WORK

In 5th generation (5G) scenarios, a considerable increase in the size and the traffic rate of wireless systems and cellular networks is expected, leading to possible problems related to scalability, reliability and latency imposed to end users. In this case, considering the use of traditional Software Defined Networking (SDN), a single controller will most probably represent a bottleneck and thus not constitute a viable solution. Moreover, the inter-connection of many switches will increase flow setup time (t_s), leading to failures on systems with Quality of Service (QoS) requirements [1]. In this sense, the control plane distributed among several SDN controllers located in different regions of the network is being studied in the literature. There are several issues to consider, such as the SDN controller location, the number of controllers required, and the assignment of switches to the controllers.

For the Controller Placement Problem (CPP) solution we present an approach that integrates radio resource allocation and Device-to-Device (D2D) communications control. Therefore, this chapter presents the state of the art of the two main problems considered in the proposed approach, considering D2D communications:

(i) the controller placement problem; and

(ii) radio resource allocation in cellular networks.

In D2D shared or underlay mode, an efficient use of the radio spectrum leads to efficient techniques for the reuse of radio resources. D2D links must reuse the RBs assigned to CUEs, so that interference occurs within the QoS limits accepted by the CUEs. The literature reports techniques that minimize interference through power control and Radio Resource Allocation (RRA).

On the other hand, in recent years, the idea of convergence between D2D communications and social networks has drawn increasing attention from the scientific community, which has started to apply resources from social networks for improving the performance of D2D communication. The rapid success and wide dissemination of social networks through platforms such as Twitter and Facebook have generated large amounts of data on the structure and dynamics of social networks, encouraging researchers from various areas to apply them to networks.

5G cellular networks are expected to support a number of emerging applications with varying QoS requirements, leading to services categorized as Enhanced Mobile Broadband (eMBB), Massive Machine Type Communication (mMTC), and Ultra-reliable and Low Latency Communications (URLLC) [28]. A literature review on the resource allocation problem considering URLLC services is provided at the end of this section.

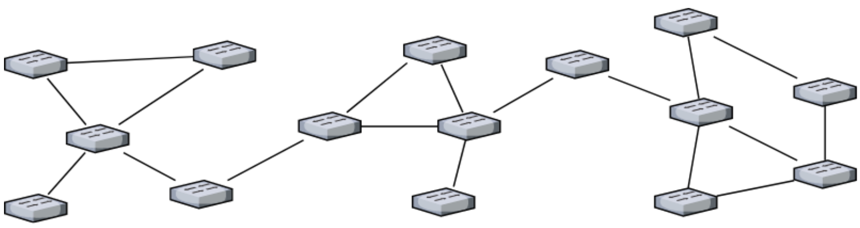
Section 2.1 and 2.2 are focused on categorizing and summarizing the most pertinent literature related to CPP and RRA problems, respectively. Section 2.1 provides, first a brief introduction to the CPP problem, and then summarizes related works, while Section 2.2 presents, first the categorization on the underlaying D2D radio resource allocation considering full and partial Channel State Information (CSI), and then on the benefits of social networks for radio resources allocation.

2.1 CONTROLLER PLACEMENT PROBLEM

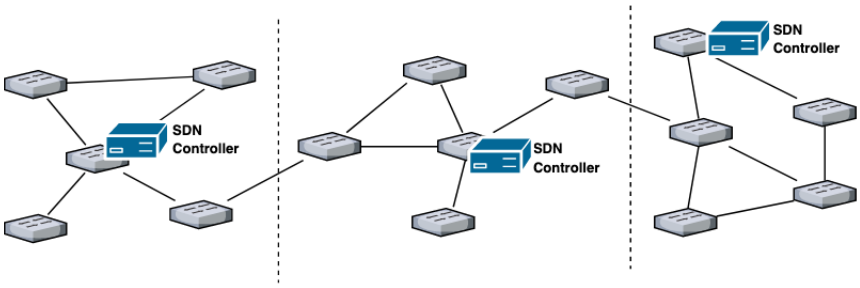
In large-scale cellular networks, multiple controllers need to be deployed for D2D-capable User Equipment (DUE) to numerous reasons: latency, load capacity of controllers, scalability, incremental deployment, and reliability. Therefore, for the design of an efficient control plan, the Controller Placement Problem (CPP) considers mainly three issues (Figure 2.1):

- the number of controllers to be placed in a network;
- the location of these controllers; and
- the function of assigning controllers to switches.

In general, CPP is modeled as an optimization problem that considers objectives, such as shortening latency, increasing reliability, increasing energy efficiency, and so on.



(a) Network Topology



(b) CPP solution

Figure 2.1: Example of Controller Placement Problem with three controllers.

Heller et al. [32] initiated the controller location problem and formulated it as a general facility location problem. The authors sought for the best location of k controllers on a WAN network by minimizing the propagation latency between the nodes and the controller. The authors evaluated the propagation latency with the use of concepts of average-case latency and worst-case latency in the solution of the optimization problem. The number of controllers was set in the interval $[1, 5]$

and the impact of the controller location on the two metrics was evaluated independently. The Internet2 OS3E topology was used and the analysis was expanded according to several topologies defined in Internet Topology Zoo (a controller was usually sufficient). However, important metrics in the evaluation of the response time for satisfying a given QoS, such as processing capacity of the controllers and load balancing, were not considered. The authors pointed out, as an extension of the work, the selection of the controller, which defines the switch-controller assignment.

Hu et al. [33] maximized the reliability of the network control to reach a best location for the controllers that minimizes the loss of the control path caused by network failures. A reliability metric represented the percentage of paths broken DUE to network failures. The problem was defined as NP-hard with k controllers and random placement, greedy search, simulated annealing and brute force were used to solve it. The authors evaluated the impact of the number of controllers on the reliability of the network and the tradeoffs between reliability and latency. The results showed latency increases in locations that optimize reliability; however, according to the authors, this increase is still acceptable. Metrics related to response time and QoS assurance were not optimized.

Sahoo et al. [34] used metaheuristics based on biologically inspired populations to solve the CPP problem in large-scale WAN networks. They applied Particle Swarm Optimization (PSO) and Firefly Algorithm (FA) to find the optimum location of the controllers and minimize latency between switches and controller. They compared the two meta-heuristics in analyses with a random localization strategy in three topologies defined on the Topology Zoo website and, according to the results, FA achieved better performance.

The main metrics considered in the resolution of the controller location problem are propagation latency between switches and controllers, and load balancing between controllers. However, the capacity of the controllers and the load on the network are also important metrics owing to their direct influence on the response time of the controllers, hence, users' quality experience. The following articles deal the optimization of the SDN controllers assignment considering their processing capacity.

Farshin and Sharifian [1, 35] optimized the number of SDN controllers used in a cellular network. They considered the problem of dynamic allocation of controllers and proposed a framework that uses a population-based metaheuristic to solve it. In [35], the authors consider cloud data centers with one Global Controller (GC) and multiple Local Controller (LC). In [1], it is considered an architecture called Software Defined Mobile Networking proposed by the authors that uses cloud computing, SDN, and Network Function Virtualization (NFV). In the above articles, the controller assignment problem was modeled as an optimization problem that maximizes the weighted sum of three utility functions that considering the response time of the controllers, the number of controllers required, and the load balancing between controllers using the Jain's Fairness Index.

To solve the optimization problem in [35], the authors use the Particle Swarm Optimization Gravitational Search Algorithm (PSOGSA) and Whale Optimization Algorithm (WOA) to implement the Controller Provisioning Algorithm (CPA), which runs in the GC and Switch Assignment Algorithm (SAA), which runs in each LC. The CPA assigns an integer number to each class of service in the datacenter for the required number of controllers, and SAA finds the best switch assignment for controllers provided to each class by CPA. In [1], it is used a chaotic population-based meta-heuristic in the framework for solving the Dynamic Controller Assignment or Allocation Problem (DCAP). However, the location of the SDN controllers in the infrastructure was not taken into consideration.

Wang et al. [36, 37] address the DCAP as a stable matching problem with transfers. In [36], the authors formulate the DCAP as an optimization problem aiming at minimizing the controller response time and control traffic overhead. In [37], the problem is formulated as an online optimization to minimize the total cost caused by response time and maintenance on the cluster of controllers. To solve the problem, they proposed a two-phase algorithm. First, a stable matching is generated between switches and controllers, which guarantees the response time in the worst case. It serves as an input to the second coalition game phase to further reduce the response time. The second phase of the algorithm is essential for load balancing, and can increase control traffic overhead due to the migration of the switch from one controller to another. The objective is to decide the number of active controllers and the proper assignment between switches and controllers.

Cello et al. [38] proposed an SDN switch migration mechanism, called Balanced Controllers (BalCon), able to achieve load balance among SDN controllers. BalCon is a heuristic that operates during the network runtime and can detect and solve congestion at the SDN controllers through the SDN switch migrations. The authors consider a fixed number of controllers, which may lead to idle controllers on the network at times of low load.

Filali et al. [39] proposed, in a data center environment, a dynamic assignment of switches to controllers by formulating the problem as a one-to-many matching game with a minimum quota that each controller has to achieve. This quota represents the utilization of the processing capacity of the controller. The authors formulated the problem as an optimization problem to minimize the total response time of the control plane. Then, to solve Controller Assignment or Allocation Problem (CAP), a one-to-many matching game with minimum resource utilization was defined, in which each controller must reach its minimum quota concerning the transmitted requests by the switches. To achieve a balanced load, the authors set capacity and minimum resource utilization constraints that controllers must approve when receiving requests from switches. The authors fix the number of controllers, and the algorithm was centered on distributing the load among the SDN controllers. This result may lead to the use of unnecessary controllers on the network when the load is low, and many controllers would be operating under load, while fewer controllers could manage all flows.

Savas et al. [40] study the CAP problem considering connectivity recovery after network failures. The authors proposed an algorithm for recovery-aware switch-controller assignment and routing (RASCAR), which enables data-path recovery after a set of failures. They formulated the problem as an integer linear program and proposed a heuristic to solve large problem instances, called Neighbor-Disjoint Path Routing. RASCAR approach determines controller assignment and control-path routing by minimizing the number of control-path recovery stages after a failure. In [40], the processing capacity of SDN controllers is not considered, so if a more significant amount of switches need to be assigned to a specific controller, after a failure, its capacity could be exceeded.

Suh and Pack [41] formulated a master controller assignment problem to minimize the average flow setup latency with consideration of the number of assigned master controllers along a path. The author first derives the load on the controller and the average flow setup latency with consideration of the number of master controllers along a path and the flow arrival rate among switches. After that, they formulated a problem as an Integer Non-Linear Programming (INLP) problem and proposed a heuristic called Low Complexity Master Assignment (LCMA) to solve the problem. In LCMA, a specific number of switches are assigned to the controllers using the brute-force search (BFS) algorithm. In contrast, to the remaining switches, the nearest controller is designated as their master controller.

Zhang et al. [42] presented an adaptive controller assignment (ACA) scheme for multiple controllers that dynamically adjust the number of active controllers and the mapping relationship among controllers and switches. The authors assume that controllers have already been deployed to the proper locations in the network. Three algorithms were proposed. First, when some controllers are overloaded, that is, the load ratio of controllers reaches a certain level, other controllers are activated to share the load of these controllers. Two, when some controllers are lightly loaded, these controllers are deactivated to reduce the control traffic overhead and improve controller utilization. Three, to further enhance network performance, the load among controllers is balanced after reassignment. In [42], authors formulated the Dynamic Controller Assignment or Allocation (DCA) problem as an integer linear program. The objective was to find an assignment that minimizes the flow setup time and control traffic overhead in the network. The authors used the random multiple-mapping matrix among controllers and switches to generate the initial assignment solution. In the approach, if there are some switches not assigned to any controller, they will be randomly connected to one of the activated controllers, but this may violate the restrictions of controller capacity. However, the authors claim that this particular case rarely happened in their simulations.

Hock et al. [43] modeled CPP as a multi-objective problem and proposed a framework called Pareto-based Optimal COntroller-placement (POCO). They obtained the best location of k controllers considering first all possible combinations of faults of up to $k-1$ controllers and then evaluating disruptions in the network. The metrics used were maximum node-controller latency and controller-controller latency in scenarios with failures, and load imbalance between controllers. The authors evaluated Topology Zoo topologies and the Internet2 OS3E topology, and showed the best latency value and best resiliency or fault tolerance cannot be obtained simultaneously. Therefore, the metrics involved must be compensated for, depending on their importance in the goal pursued. The authors explored the search space for a small number of controllers exhaustively; however, in large-scale networks, the obtain of such results in acceptable time and with the existing computing capabilities is a difficult task.

Liao and Leung [44] also modeled CPP as a multi-objective problem. The authors proposed Multi-Objective Genetic Algorithm (MOGA) for the solution of the CPP problem. The mutation function is based on PSO. The authors presented optimal Pareto solutions by minimizing switch-controller latency, controller-controller latency, and load imbalance between controllers. In the tests performed, they defined the number of controllers and sought for the location and assignment of the switches.

Ahmadi and Khorramizadeh [45] adapted multi-objective genetic algorithms to solve CPP and introduced Multi-start hybrid non-dominated sorting genetic algorithm (MHNSGA). They evaluated two approaches to the problem. The first considers the minimization of the following metrics: inter-controller latency, link or switch failures, and load balancing between controllers (using the load imbalance concept), whereas the second includes both load and capacity of the controllers in the evaluation metrics as a constraint of the problem. Solutions are sought for near the Pareto front and the trade-off between the metrics is conducted. The number of controllers at work must be known and several topologies defined in Internet Topology Zoo are used in the evaluation.

Bari et al. [46] presented a solution that dynamically adapts the number of controllers and their location to the conditions of the network. They adjusted the number of active controllers and assigned OpenFlow switches dynamically minimizing flow setup time and communication overhead. Moreover, they considered the statistics collection cost, flow setup cost, synchronization cost and switch reassignment cost, which represent, respectively, number of messages necessary for the collection of network statistics, cost of the path regarding propagation latency or number of hops, average number of flows generated by the switches, number of messages exchanged between the controllers, and cost of reassignment of a switch to a new controller. The authors used a simple modeling of the processing capacity of the controllers, which was defined in a vector representing the maximum number of requests each controller can manage per second.

Yuan et al. [47] formulated the SDN controller assignment problem as a minimum weight matching of the bipartite graph. In [47], the Kuhn-Munkres algorithm based solution was used to find optimal matching between switches and controllers. Then, a genetic algorithm was proposed to solve the controller placement problem based on the controller assignment scheme. To define the weights considered in the bipartite graph, the authors used the propagation delay between controllers and switches as a metric. The proposed approach considered load balancing between the controllers based on the total capacity of the SDN controllers. The capacity of controllers was defined as the maximum number of switches that the controller can serve. The article [47] did not consider network changes that lead to different request rates received by SDN controllers, neither the flow setup time.

Rath et al. [48] proposed a solution that dynamically enables/disables controllers considering the load on the network. According to the optimization mechanism used, each controller calculates a payoff function and compares its own value with that of neighboring controllers. It then makes appropriate decisions, so that new controllers are added or existing ones are deleted, or the download is performed between the controllers dynamically. The authors considered delay and usage constraints, such that the delay associated with the controller must be lower than a predefined threshold value (to support QoS) and the utilization must be within the minimum and maximum limits. Delay combines path and processing latency. The purpose of the scheme is to minimize packet loss, latency, and deployment cost.

He et al. [49] analyzes the controller placement for dynamic traffic flows based on a combined controller placement model: controller locations and switch-to-controller assignments are simultaneously optimized for minimum average flow setup time with respect to different traffic conditions inside the network. Linearization method is applied to transform the problem into a Mixed Integer Programming (MIP) problem, and the Gurobi optimizer is used in the solution. The authors present the Controller (CTR) model and the Switch (SW) model. In the CTR model, the control domains remain unchanged, while the controller moves inside their control domains. SW model targets the scenario that controllers stay in their places, and switches change their assignments. In that article, the authors do not determine the number of controllers needed to manage flows, nor do they consider load balancing between them. Therefore, the results may contemplate the use of unneeded controllers or with a considerable difference between their load (there may be low load controllers in the network while others have a high load).

Abdel-Rahman et al. [50] introduced a framework to solve the CPP problem in SDN cellular networks involving the uncertainty of the geographical distribution of users. Two schemes were proposed, of which one aims to optimize the number of controllers necessary for the management of all evolved NodesB (eNBs) while ensuring the response time of each eNB does not exceed a specific value with a certain probability. The other scheme aims to optimize the eNB-controller assignment considering the variation in the eNB request rate for minimizing the response time for several eNBs. The eNB-controller assignment adapts to fluctuations in the eNB request rates, resulting from the variations in the cellular user locations. The authors derive an equivalent Mixed Integer Linear Programming (MILP) formulation, and it was using CPLEX to solve the problems. In the article, the authors do not consider load balancing between controllers.

Considering the metrics used in the literature, we investigated the CPP problem using the average response time of the SDN controllers and the eNB-controller latency within the response time. We modeled the processing of the controllers applying the queuing theory as in [1, 50, 41, 42]; moreover, we considered the request rate received by the SDN controllers or control traffic generated by the D2D communications established in the eNB coverage area. Solving the problem, we used the CPP Artificial Bee Colony (CPP-ABC) and CPP Ant Colony System with External Memory (CPP-ACS-EM) metaheuristics to find an approximate solution in an acceptable time. The path taken by ants and the bees food source contribute to the determination of controllers number, the eNB-controller assignment, and SDN controllers location.

Table 2.1 shows a summary of the main characteristics of the studies on the CPP problem.

Table 2.1: Strategies used in CPP.

Proposal	Details	Metrics	Strategies	Focus on D2D Communications
Heller et al. [32] (2012)	best location by minimizing the node-controller propagation latency	average-case latency; worst-case latency	minimum k- median (local search heuristic for k-median); minimum k-center	No
Hu et al. [33] (2013)	best location of controllers that maximizes network reliability	reliability	random placement; l-w-greedy; simulated annealing; brute force	No
Sahoo et al. [34] (2017)	best location for controllers that optimizes latency	node-controller latency	PSO; FA	No
Farshin and Sharifian [1, 35] (2017)	dynamic calculation of number of SDN controllers allocated considering network traffic	controller response time; controller utilization; Jain's Fairness Index	Grey Wolf Optimizer (GWO); PSOGSA; WOA	No
Wang et al. [36, 37] (2016, 2017)	dynamic controller assignment, the switches are dynamically mapped to different controllers when traffic varies	controller response time; control traffic overhead	Stable matching problem with transfers; Coalitional game theory	No
Cello et al. [38] (2017)	SDN switch migration mechanism that operates during the network runtime and can detect and solve congestion at the SDN controllers	load balance	BalCon heuristic	No
Filali et al. [39] (2018)	dynamic assignment of switches to controllers in a data center environment	total response time; resource utilization and capacity constraints	one-to-many matching game	No
Savas et al. [40] (2018)	assignment problem considering connectivity recovery after network failures	number of control-path recovery stages	Neighbor-Disjoint Path Routing heuristic	No
Suh and Pack [41] (2018)	SDN master controller assignment problem	average flow setup latency; controller load	INLP; heuristic	No
Zhang et al. [42] (2018)	adaptive controller assignment scheme for multiple controllers that dynamically adjust the number of active controllers and the mapping relationship among controllers and switches	flow setup time; control traffic overhead; load balance	ACA scheme, CAP model as knapsack problem	No
Hock et al. [43] (2013)	better location considering higher resilience and fault tolerance	node-controller latency; controller-controller latency; load imbalance	Pareto-based optimal placement (considering all solution space)	No
Liao and Leung [44] (2017)	best location and assignment of switches minimizing latency and load unbalance between the controllers	node-controller latency; controller-controller latency; load imbalance	MOGA with a PSO-based mutation	No

Table 2.1: Strategies used in CPP.

Proposal	**Details**	**Metrics**	**Strategies**	**Focus on D2D Communications**
Ahmadi and Khorramizadeh [45] (2017)	better location and assignment of nodes to controllers, minimizing latency and load unbalance between controllers with load restrictions	node-controller latency; controller-controller latency; load imbalance; capacity of controllers; switch load	MHNSGA	No
Bari et al. [46] (2013)	number of active controllers and assignment of switches to controllers considering network failures	statistics collection cost; flow setup cost; synchronization cost; switch reassignment cost	a greedy approach based on the knap-sack problem; a simulated annealing-based meta-heuristic approach	No
Yuan et al. [47] (2018)	controller placement problem based on the controller assignment scheme	propagation delay; load balance, controller capacity	Kuhn-Munkres algorithm (bipartite graph); genetic algorithm	No
Rath et al. [48] (2014)	best assignment and location of controllers after increasing/decreasing the number of controllers according to the load on the network	latency (processing and path delay); controller utilization	Non-zero-sum-based game theory	No
He et al. [49] (2017)	controller placement for dynamic traffic flows, controller locations and switch-to-controller assignments are simultaneously optimized	end-to-end flow setup time	MIP; Gurobi	No
Abdel-Rahman et al. [50] (2017)	minimum number of SDN controllers, their ideal locations and assignment to eNBs, where the optimization criteria are based on the satisfaction of the eNBs delay requirements	transmission and propagation latency; queuing latency controller	Stochastic programming; MILP; CPLEX	No
Our CPP proposal	number of controllers, their location and eNB-controller assignment considering response time	controller response time; eNB-controller latency; Jain's Fairness Index	CPP-ABC; CPP-ACS-EM	Yes

According to our literature review on the SDN CPP problem for wireless networks, the treatment of D2D communications has not been addressed, and the Artificial Bee Colony (ABC) and Ant Colony System with External Memory (ACS-EM) meta-heuristics have not been applied to the problem considered here. The first solution proposed considers two steps and two subproblems. In the first, the radio resource allocation problem is solved by a greedy approach, according to which the number of D2D communications accepted in the network is obtained and used as input to the CPP. In the second step, the CPP problem is solved according to the aforementioned description (our CPP proposal).

The study of radio resource management is deepened in the sequence. The next section analyzes research related to the radio resource allocation problem in 5G cellular networks.

2.2 RADIO RESOURCE ALLOCATION

Radio Resource Management (RRM) is the control of features and configurations of wireless communication systems such as cellular networks, broadcast systems and wireless data networks. The RRM involves from the power transmission control, channel allocation, bandwidth and error correction schemes, to more complex operations such as planning the reuse of frequencies, and load balancing. Of all the operations and controls considered in RRM, in this thesis we will be studying those related to Radio Resource Allocation (RRA) such as transmit power control, channel allocation, communication mode and resource reuse.

From the RRA perspective, D2D communication can be classified by several criteria depending on the reuse of radio resources, the used communication mode, the type of interference management, and the power control (Figure 2.2). Radio resources in mobile networks are originally allocated for Uplink (UL) and Downlink (DL). D2D communications, depending on the communication mode, may reuse the UL, the DL, or both. The most common approach in the current literature is to use the UL resources of cellular networks. The advantage of the UL is that this direction is mostly underutilized compared with the DL, since most users would rather download data from the network. In addition, the interference situation in the UL is much easier to resolve with respect to cellular transmission because the victim of D2D interference is solely the eNB.

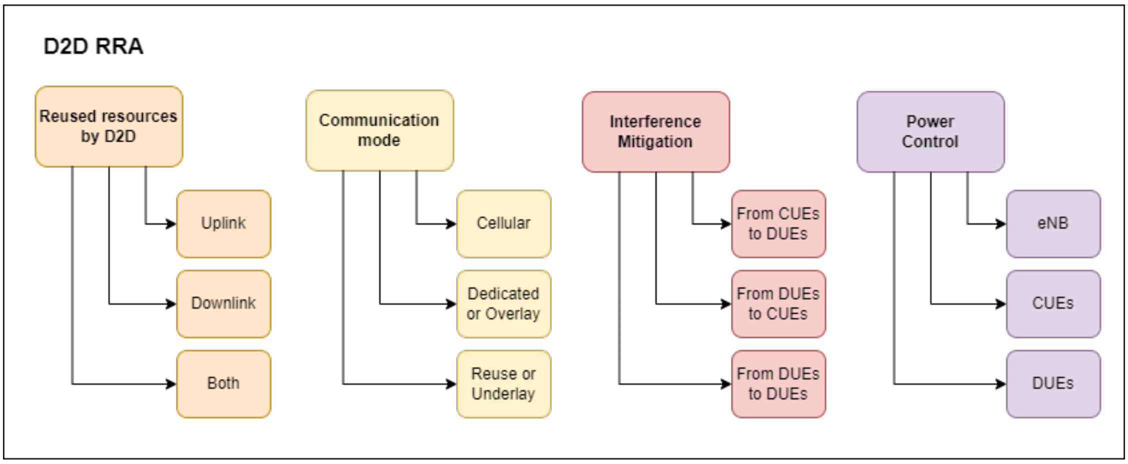
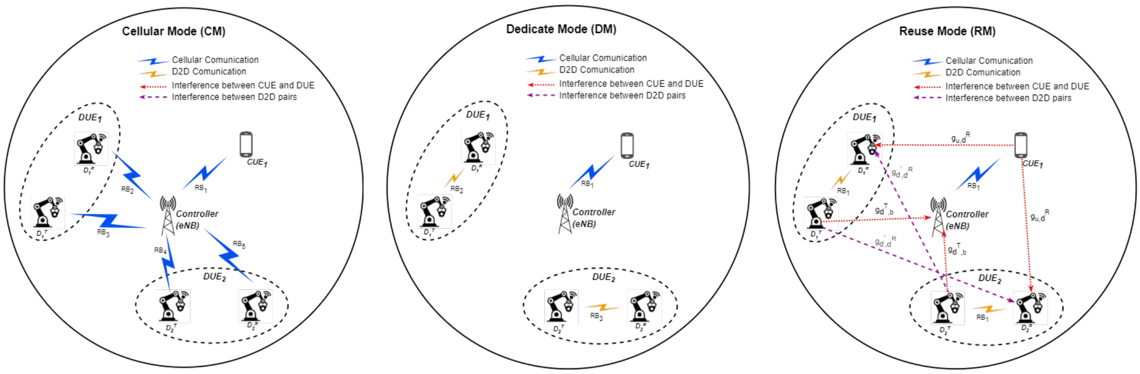


Figure 2.2: D2D communication from RRA perspective

The communication mode identifies whether the DUEs communicate directly with each other or via the eNB. It distinguishes if D2D communication uses the same radio resources as the conventional cellular communication or not. The literature recognizes three communication mode:

1. Cellular Mode (CM): is conventional cellular communication, the DUEs exchange data through the eNB and no direct exchange of data between the DUEs takes place. As shown



(a) Cellular Mode (CM) (b) Dedicated Mode (DM) (c) Reuse Mode (RM)

Figure 2.3: Examples of D2D communication modes

in the Figure 2.3(a), a free resource block (RB) is assigned to the transmitter and the receiver of D2D communication;

2. Dedicated Mode (DM) or overlay mode: is a mode allowing two DUEs to transmit data directly between themselves without intermediate eNB, the eNB has to dedicate radio resources for the DUEs' transmission. As shown in the Figure 2.3(b), a free RB is assigned to D2D communication;

3. Reuse Mode (RM) or underlay mode: a same radio resources are used both for the DUEs and for the cellular User Equipments (CUEs). As shown in the Figure 2.3(c), a RB assigned to cellular communication is used by D2D communications.

From the spectral efficient perspective, CM has the lowest efficiency because direct communication is not used. DM has a medium efficiency, where the eNB has to dedicate radio resources for the DUEs' transmission and, thus, the CUEs cannot exploit the full capacity of the eNB. In DM, the radio resources are used with a higher efficiency than in the case of the CM, since only one transmission direction, either the DL or the UL, is used for the D2D transmission. Similarly to the DM, the RM can use either the DL or the UL radio resources. In RM the reuse factor is significantly higher than in the case of the CM or the DM, therefore, this mode is even more profitable for the system.

From the interference perspective, the advantage of the DM is that the eNB does not need to handle interference among the CUEs and the DUEs. In this mode, transmission of the CUEs and the DUEs has assigned a non-overlapping orthogonal radio resource. In CM, the interference can be easily managed by the eNB and no new features have to be implemented. However, in RM, strong interference could be generated among the DUEs and the CUEs. To prevent the generation of harmful interference, new techniques and procedures have to be introduced to the system. As a result, the complexity of whole system is increased.

The cellular communication should not be affected by the introduction of D2D communication, which makes interference a critical problem in D2D scenario. In addition, if the D2D communication is strongly interfered by the cellular communication, its applicability and/or efficiency is significantly reduced. The nodes that are affected by interference (eNB, CUE, DUE) depend on D2D communication mode and on the resources used for D2D communication (UL/DL). Interference between the CUEs and the DUEs occurs only if the DUEs are in the RM. Furthermore, interference between D2D pairs is introduced only if the same resources are allocated to them.

An interference classification was presented in [51], considering who is the interferer and who is the victim of interference, as follows:

1. Interference from the D2D communication to the cellular communication: in the UL direction, the interference is caused to the eNB, which receives data from its CUEs. In the DL direction, the CUEs are the victims of the D2D interference as they are receiving data from the eNB at the same time as the DUEs and exchange data among themselves.
2. Interference from the cellular communication to the D2D Communication: in the UL direction the interference to the D2D communication is generated by the CUE, which transmits to the eNB. In the DL direction, the source of interference is the eNB.
3. Interference between D2D pairs: if more than one D2D pair is reusing the same radio resources, the additional concern is the interference generated between the DUEs.

2.2.1 Radio resource allocation problem in D2D communications.

In a D2D communication system, when D2D pairs simultaneously share the same spectral resource with the CUE, adjacent D2D pairs and the CUE may cause interference with each other. Therefore, interference and radio resource allocation are crucial issues addressed in research worldwide.

Sun et al. [52] formulated the uplink resource allocation problem where more than one D2D pair can share the same resource with one CUE under the constraint that the Signal-to-Interference-plus-Noise Ratio (SINR) requirements of CUEs and admitted D2D pairs are satisfied. The mutual interferences among different D2D pairs as well as that between DUE and CUE sharing the same RB resource was considered by authors. The resource allocation problem was formulated to maximize the number of admitted D2D pairs by enabling more D2D pairs to reuse the same RB resource subject to the constraint of the maximum tolerable interference level at each RB. The authors proposed a greedy resource allocation heuristic based on the smallest degree interference criterion from the graph theory.

Ciou et al. [53] proposed the Greedy Throughput Maximization Plus (GTM+) algorithm to solve the multi-sharing resource allocation problem. GTM+ exploits conflict graph and maximal weight independent set to improve system throughput while ensuring the minimum SINR requirements of CUEs and DUEs.

Esmat et al. [54] designed two phases-based resource sharing algorithms. The initial set of candidate channels that can be reused by each DUE is adaptively determined in the first phase.

In the second phase, Lagrangian dual decomposition is used to determine the optimal power for DUEs that maximizes the network sum-rate. The authors considered that multiple DUEs are allowed to reuse the same CUE channel, each DUE can use multiple CUEs channels and adaptive power control for DUEs.

Wang et al. [55] formulated a problem of maximization of ergodic sum rate of D2D underlay system subject to both D2D and cellular outage constraints, based only on statistical (partial) CSI. The authors constructed a simplified bipartite graph for resource sharing between cellular links and D2D links.

Kai et al. [56] investigated the joint uplink and downlink resources allocation to maximize the sum data rate for all users in a system, ensuring QoS for CUEs and DUEs. They designed a Joint Uplink and Downlink resource allocation scheme composed of two steps. First, a power control scheme was developed for each possible DUE-CUE pair to maximize the overall data rate, and then, a maximum weight bipartite matching scheme was designed to assign a subcarrier to each DUE. The authors assumed the Base Station had the perfect CSI of all links involved for designing a centralized resource allocation scheme.

Ali et al. [57] proposed the Termed Threshold Controlled Access algorithm for uplink resource allocation based on battery status and related application's power profile that eventually leads to required QoS metric. Initially, the algorithm selects the number of carriers to be allocated to a certain device for the better lifetime of low power devices. Then, a threshold is obtained based on the QoS metric for the selection of sub-carriers for less powered devices, such as small e-health sensors. The results show that this approach achieves a great performance.

Dominic and Jacob [58] proposed an interference-aware admission and pricing-based power control tools to ensure CUEs data rate above a minimum and interference below a limit respect to the D2D pairs. The algorithm maximizes the sum rate of the D2D tier while meeting QoS constraints of the CUEs and the D2D pairs. As a differential, the authors introduce an interference limit as a constraint for the D2D pairs.

Gao et al. [59] presented a cooperative model and then consider the total performance problem for the entire network with the premise of guaranteeing the quality of the CUEs communication. The authors implemented the quantum coral reefs optimization algorithm for the joint allocation of resources and the problem of power control in heterogeneous D2D networks, whose objective is to maximize the overall performance.

Lee et al. [60] give a different approach to the resource allocation problem. The authors introduce a Machine Leaning (ML) technique rather than formulate the problem as mixed integer nonlinear programming. The main idea is to learn a prune policy to enhance the branch-and-bound algorithm. Finally, the problem is modeled as a binary classification problem and solved by the training of a support vector machine model.

Gorantla and Mehta [16] studied sub-channel allocation to D2D pairs in the underlay mode in a multi-cell scenario with multiple uplink sub-channels and limited CSI. They proposed a q-bit quantized feedback scheme in which a D2D pair feeds back its

Kwon and Kim [61] presented a three-dimensional channel assignment with pairs of (CUE, DUE, channel) considering a system whose channel bandwidth is not fixed, but adaptively determined to according to triple CU, DU, channel. They proposed a sub-optimal algorithm that calculates power in a closed form using Lagrangian relaxation and a sub-gradient-based iterative algorithm. The CSI of all possible combinations of links was assumed available at the BS, which might require indispensable CSI feedback.

2.2.2 Social-aware radio resource allocation in D2D communications.

Social networking concepts can help find solutions and enhancements for D2D communications in both physical domain, where devices can help or be assisted before accessing a network or initiating D2D communication, and social domain, where devices can create a social network governed by social relationships and events. Considering such two main domains, we used information of social interaction between D2D users in the radio resource allocation, where D2D communications employ the same spectrum resource as cellular communications.

The topic of resource allocation has also been analyzed since the beginning of the social awareness approach. Below are studies that use information of social relations among users in the radio resources allocation problem.

Wang et al. [62] modelled the sociality of D2D transmission links in terms of contact time as complementary indicator of the link strength and formulated both sociality-non-aware and sociality-aware optimisation problems for the resource allocation in D2D-assisted cellular networks. The research focused on the maximization of resource utilisation and network utility but satisfying congestion constraints. The authors demonstrated that the sociality-aware resource allocation can achieve higher performance than that of the sociality-non-aware approach.

Feng et al. [63] considered the similar and in common characteristics of people in small social communities and exploit them to optimize the resource allocation in the communities region. The RAA is solved by finding the optimal graph matching which derived via bipartite graph matching. Finally, the authors developed a low-complexity small social community resource allocation algorithm to solve the problem.

Deng et al. [64] optimized the resource allocation of D2D communication by utilizing the social relations embedded in the communication devices. The authors used the call records to quantify the social strength by the application of the auto-regressive integrated moving average model. Then, users maximize its social-community utility considering the other social related D2D users. Using potential game it is solved the social-community utility maximization problem of resource allocation for D2D communication and a social-aware distributed resource allocation algorithm is proposed.

Huynh et al. [65] proposed another social-aware and energy efficiency optimization solution for D2D communications in 5G networks. First its is analyzed and evaluated the influence of social factors on the performance of D2D communications. Moreover, the authors define and solve an energy efficiency optimization problem considering both the social relationships and physical interference between all the User Equipments (UEs). The problem considers an optimal channel mode selection and optimal transmission powers allocated to each UE, maximizing the energy efficiency, by utilizing adaptive genetic algorithm.

Yang et al. [66] designed a QoS-driven social-aware network model to integrate social information into cognitive D2D-based Internet of Things (IoT) networks. The authors modeled a social-aware energy-efficient resource management optimization problem as a multiagent reinforcement learning formulation and developed a coordinated multiagent deep reinforcement learning (DRL)-prioritized experience replay (PER)-based resource management approach to optimize both joint radio block assignment and transmission power control strategy.

Our first RRA proposal allocates radio resources aware of the intensity of social relations between users. CUEs use orthogonal channels, so that interference between them is not considered. The conflict graph is used for each CUEs for the modeling of interference between D2D pairs that can use the same RB assigned to a CUE. The greedy heuristic (Social-aware Greedy Resource Allocation Algorithm (SA-GRAA)) solved the problem selecting the D2D pairs with the highest social relationship. The proposed solution verifies if the QoS parameters of both CUEs and DUEs complain of all validity interference criteria.

The resource allocation problem in which more than one D2D pair can share a same RB with a piece of CUE is investigated. RBs are assigned to CUEs and the QoS of both CUEs and DUEs is considered. Mutual interferences among different D2D pairs, as well as that between DUE and CUE sharing the same RB are taken into account. Moreover, the achievement of high spectral efficiency while ensuring no excessive interference is caused to CUEs depends on an adequate interference-aware allocation of sub-channels to D2D users. In this scenario, eNB must have full or partial CSI to perform an efficient resource allocation.

The full CSI model was considered and, according it, eNB knows the perfect CSI of all links [16, 17, 18], whereas in practice, it knows only the channel power gains of CUEs for eNB and D2D transmitter (d_j^T) for eNB links - a priori, it does not know the channel power gains of CUE for D2D receiver (d_j^R) and d_j^T for d_j^R links. Therefore, the CSI of such links can be obtained through feedback, which causes high overhead [17, 18], especially when the number of CUEs and D2D pairs is large.

In the feedback model, the D2D receiver sends eNB information to the CUE for d_j^R and d_j^T for d_j^R channel gains. Therefore, in the second RRA solution, we propose the selected-*NM* Maximum Distance Ratio (MDR) q-bit feedback scheme which, considering the increase in network overhead generated by feedback, limits the sending of q-bits of channel information, as well as the number of links. Our second RRA proposal also allocates radio resources aware of the intensity of social relations between users and a conflict graph. However, it considers partial CSI and an ABC meta-heuristic solves the radio resource allocation problem that maximizes the network throughput.

The use of social relations as a parameter for the selection of D2D links is expected to increase the throughput on the network, since users with similar social interests can consult the same content on social networks (e.g., videos, images, among others). Therefore, if a D2D device has already consumed a content, it can distribute it to another D2D device near it. For example, if two users are fans of the same football team, they are expected to watch the same videos of their team, published on social networks, such as Twitter. If one of them has already watched the video, he can transmit it to a nearby D2D device and share the same social interest.

Table 2.2 shows a summary of the main characteristics of the studies on the D2D RRA problem.

Table 2.2: Strategies used for the solution of the D2D RRA problem.

Proposal	**Reused Resources**	**Interference Model**	**Target**	**Metrics**	**Solution and Tool**	**Social-aware**
Sun et al. [52] (2013)	Uplink	Conflict Graph	maximize the number of admitted D2D pairs	degree interference and SINR	Greedy heuristic Resource Allocation Algorithm (GRA)	No
Ciou et al. [53] (2015)	Uplink	Conflict Graph	improve system throughput	SINR	Greedy Throughput Maximization Plus (GTM+)	No
Esmat et al. [54] (2016)	Uplink	Array with maximum interference limit for each CUE	maximize the overall UL network throughput	SINR and transmit power	Adaptive Resource Sharing Algorithm	No
Wang et al. [55] (2017)	Uplink	statistical model	Maximization of the ergodic sum rate of the system under outage constraints	CSI, outage probability, SINR and data rate	Weight bipartite graph and the Hungarian algorithm	No
Kai et al. [56] (2018)	Uplink and Downlink	Not explained	Maximization of the data rate of the overall system	SINR and transmission power	Mixed integer non-linear programming problem (MINLP) and weight bipartite matching	No
Ali et al. [57] (2019)	Uplink	Not explained	Improve energy consumption	SINR and transmit power	Termed Threshold Controlled Access algorithm	No
Dominic and Jacob [58] (2019)	Downlink	SINR	Maximizes the sum rate of the D2D tier	transmit power and degree of intimacy	Game theory and Stochastic Learning Algorithm	No
Gao et al. [59] (2019)	Downlink	Array with interference between CUEs and DUEs	Maximize the total throughput, optimizing resource allocation and power control scheme	SINR	Quantum coral reefs optimization algorithm	No

Table 2.2: Strategies used for the solution of the D2D RRA problem.

Proposal	Reused Resources	Interference Model	Target	Metrics	Solution and Tool	Social-aware
Deng et al. [64] (2018)	Uplink	Conflict Graph and Aggregated Interference Function	Maximize the social-community utility	transmit power and degree of intimacy	Social-aware potential game approach	Yes
Lee et al. [60] (2019)	Uplink	Boolean {0,1} for link reuse	Maximize the spectrum use efficiency	Transmission power and SINR	Support vector machines and Branch and bound algorithm	No
Gorantla and Mehta [16] (2019)	Uplink	statistical model	Maximization of the sum of the D2D throughputs	SINR and CSI	cardinality-constrained subchannel assignment algorithm (CCSAA) and Cardinality-Constrained locally greedy algorithm (LGA) (CCLGA)	No
Kwon and Kim [61] (2020)	Uplink	Not explained	Minimization of the total channel bandwidth	SINR, data rate and transmission power	sub-optimal algorithm by using Lagrangian relaxation and a sub-gradient-based iterative algorithm	No
Wang et al. [62] (2015)	Downlink	Modelled as a constraint which considered an equation similar to SINR	Maximize resource utilization and network utility	Transmission power and Socially strength given time of a D2D communication	CPLEX	Both
Feng et al. [63] (2017)	Uplink	Boolean {0,1} for link reuse	Optimize the overall throughput of the communities	Throughput and social relationship degree	Small social community resource allocation algorithm	Yes
Deng et al. [64] (2018)	Uplink	Conflict Graph and Aggregated Interference Function	Maximization of social-community utility	Transmission power and degree of intimacy	Social-aware potential game approach	Yes
Huynh et al. [65] (2018)	Not specified	Matrix for interference between entities	Maximize the energy efficiency	Transmission power, SINR and social strength	Adaptive genetic algorithm	Yes
Yang et al. [66] (2020)	Uplink	Not explained	Maximization of the ratio of the sum data rate and the sum energy consumption	SINR, data rate and transmission power	Deep Reinforcement Learning-PER-based resource management	Yes
Greedy RRA proposal	Uplink	Conflict Graph	Maximize social relationship	SINR and strength of social relationships	SA-GRAA	Yes
Our first RRA proposal	Uplink	Conflict Graph	Maximization of weighted system throughput	SINR and strength of social relationships	Social-aware RRA Artificial Bee Colony (SA-RRA-ABC)	Yes

2.2.3 Radio resource allocation problem in wireless communication system for URLLC services.

The achievement of Ultra-reliable and Low Latency Communications (URLLC) requirements is a great challenge for 5G networks and will demand system design modifications to the current telecommunications infrastructure. In cellular networks, the efficient allocation of resources is one of the main problems to be approached; therefore, researchers have been interested in studying RRA considering URLLC services.

The block length of the channel codes in URLLC services must be short towards reductions in the transmission delay; the communication is no longer arbitrarily reliable and the decoding error probability should not be negligible. Shannon's capacity is not applicable for the characterization of the maximum achievable rate of short URLLC packets, as shown by Y. Polyanskiy et al. [19], who derived a fundamental relationship among achievable rate, decoding error probability, and packet length. Their results have been widely used in such a characterization for solving the resource allocation problem [25, 27, 67, 23, 68, 21, 30].

Singh et al. [69] considered a multi-cell, multi- user cellular-D2D communication network for URLLC services and presented two schemes, namely (1) fast allocation and (2) URLLC optimized allocation. In the former, the spectrum resource is shared between cellular and D2D communications according to the proportion of traffic load, whereas the latter considers the target rate maximization, i.e., the rate available for 99.999% of the locations, and the allocation can be directed to the ultra-low capacity region.

Ghanem et al. [25] considered a multi-user URLLC-OFDMA (orthogonal frequency-division multiple access) system and proposed a sub-optimal algorithm that performs resource allocation on the downlink for maximizing the system throughput subject to QoS constraints for all URLLC users. The algorithm obtains a local optimal solution based on successive convex approximations. The same authors in [27] presented a joint uplink-downlink resource allocation algorithm designed for multi-user URLLC-OFDMA Mobile Edge Computing (MEC) systems. The objective of the optimization problem was to minimize the total weighted power consumed by the Base Station (BS) and the users. The authors applied the branch- and-bound algorithm to find the global optimal solution by successively partitioning the non-convex feasible region and using bounds in the objective function to discard inferior partition elements.

Singh et al. [24] studied the resource allocation problem in multi-user multicarrier amplify-and-forward relay networks considering URLLC services. The objective was to maximize the energy efficiency in the network under the constraints of the decoding error probability. The authors designed an iterative algorithm to solve the modeled optimization problem, thus obtaining a near optimal solution.

Jayaweera et al. [67] considered an Elevated LiDAR system with URLLC communications and formulated the resource allocation problem to minimize the maximum decoding error probability. In a second step, they reformulated the problem towards minimizing the total energy consumption while achieving the targeted decoder error probability. The algorithm is iterative, with outputs which are optimal user-specific powers and corresponding block lengths for each user.

Nasir et al. [68] studied three different optimization problems. The first deals with joint power and bandwidth allocation considering a communication network where a single antenna node communicates with multiple end devices; the second performs beamforming design in a multiple-input-single-output (MISO) system, and the third treats power allocation under regularized zero-forcing beamforming in a multi-user MISO system.

Considering a mission-critical IoT scenario, Ren et al. in [30] formulated the resource allocation problem as an optimization one. The aim was to minimize the decoding error probability for the actuator while guaranteeing the decoding error probability for the robot, taking into account power and blocklength constraints. The problem jointly optimized blocklength and transmission power considering four transmission schemes, namely orthogonal multiple acces (OMA), on-orthogonal multiple access (NOMA), relay-assisted transmission, and C-NOMA (cooperative NOMA) transmission.

Librino and Santi [70] proposed a methodology for resource allocation based on graph theory in a dense smart factory scenario. The authors assumed no information on the channel state was available, so that the resource allocation was based on topology and the channel statistics. The Graph-Based Allocation Algorithm (GBA) proposed enabled resource allocation considering channel diversity in terms of interference level and balancing the resources among the devices. The authors added a resource sharing scheme to the allocation algorithm based on successive interference cancellations.

Librino and Santi [71] extended their previous research modifying the GBA resource allocation algorithm towards including an in-band pilot transmission of the devices and providing a channel estimation at an access point. They analyzed the trade-off between the channel state information (CSI) update frequency and the overall system performance in terms of traffic served in a dense network scenario.

Dong et al. [21] proposed a deep learning neural network to solve the resource allocation problem in 5G systems for minimizing the transmit power of a base station with multiple antennas. They considered three types of services, namely URLLC, delay-tolerant, and delay-sensitive and developed an optimization algorithm to solve the Mixed Integer Non-Linear Programming (MINLP) problem and obtain the training samples. Deep transfer learning tuned the pre-trained Neural Networks (NNs).

2.2.3.1 Radio resource allocation problem in D2D communications for URLLC services.

The focus of this sub-section involves the RRA problem in a scenario that considers URLLC services with the use of D2D communications, which might improve spectral utilization enabling direct communications between devices. D2D communications represent a promising technology for the achievement of high data rates and limited delay, and D2D networks reduce latency while ensuring data rate, helping to guarantee requirements of URLLC services.

Chu et al. [20] considered a cellular network where D2D communications use URLLC services and reuse the cellular link in the transmission of short packets. The authors formulated the allocation problem towards maximizing the achievable rate of a D2D pair subject to a received-power outage constraint at the cellular BS and the imperfect CSI between the D2D user and the cellular BS. They adopted the successive convex approximation method to linearize the objective function and then solve the problem with an iterative algorithm.

Yang et al. [22] derived the lower bound of the tradeoff among spectral efficiency, transmission latency, and reliability according to the finite block length theory. Based on the worst-case lower bounds, the authors studied the optimal frame design towards minimizing the transmission latency while ensuring the amount of transmission information for each V2V (vehicular to vehicular) pair and the

Sanusi et al. [28] jointly considered admission and power control to determine the feasible sets of CUE-DUE pairs that could meet the minimum QoS requirements for an industrial factory scenario. The authors proposed a strategy to mitigate co-channel interference from CUEs and DUEs based on distance so that user terminals which produce large interference are avoided. First, the set of power pairs that satisfied the distance metric for potential reuse partners was obtained. The set of power pairs that guaranteed the reliability requirement of DUEs was then acquired, and finally, the optimal power assignment that maximized the sum of the CUE and DUE output was achieved. A matching game solution that used the price-DA algorithm with an incentive- based stability to improve resource sharing was proposed for solving the problem.

I. O. Sanusi, K. M. Nasr, and K. Moessner [29] presented an optimization scheme for a cellular network with D2D in an Industrial Internet of Things (IIoT) environment which maximized the throughput of the supported D2D links without compromising the QoS of the network users. They adopted an optimization approach to solve the allocation problem.

Considering D2D vehicular networks, Yang et al. [72] introduced a joint mode selection, a resource block (RB) assignment, and a transmission power control scheme. The objective of the modeled optimization problem was to maximize the overall network capacity. The decision-making problem was formulated as a decentralized Reinforcement Learning (RL) framework. In [31], the authors presented an Energy Efficiency (EE) resource management in a heterogeneous radio frequency (RF)/visible light communication industrial network architecture, guaranteeing the requirements (high reliability, low latency, and high data rate) of IIoT and IoT devices. They proposed a deep post-decision state (PDS)-based experience replay and transfer (PDS-ERT) RL algorithm for the problem resolution towards maximizing the network EE.

Wu et al. [73] studied a contention-based access scheme to deal with sporadic packet arrival generated by D2D communications. They modeled an optimization problem to perform radio resource management in D2D communications considering URLLC services. The problem was transformed into a mixed strategy non-cooperative game towards the distributed allocation of RBs, maximizing the sum of Successful Transmission Probability (STP) of all D2D pairs.

Our second RRA proposal considers a D2D cellular network for an industrial factory scenario where D2D communications use URLLC services and assumes D2D pairs can reuse cellular links if channels are not available. Therefore, different communication modes for D2D pairs were considered for the study of the resource allocation problem, formulated as an optimization problem for maximizing the energy efficiency subject to the SINR and maximum power constraints of CUEs and D2D pairs. Three bio-inspired ABC-based, ACS-EM-based, and PSO-based metaheuristics were proposed for the obtaining of near-optimal solutions.

Table 2.3 shows a summary of the main characteristics of the studies on RRA problem in wireless communication systems for URLLC services.

Table 2.3: Strategies used in wireless communication system for URLLC services.

Proposal	**Focus**	**Objective**	**Techniques / Algorithms**	**D2D**	**Scenario**
Singh et al. [69] (2017)	Uplink and Downlink	Maximization of the minimum achievable rates	Estimation of transmission rates and protocol for channel allocation based on the estimated values	Yes	Industrial (Factory)
Ghanem et al. [25] (2019)	Downlink	Maximization of the system throughput	big-M formulation, Integer relaxation and Difference of convex programming	No	Cellular network
Ghanem et al. [27] (2020)	Uplink and Downlink	Minimization of the total weighted power consumed by both BS and users	Branch-and-bound Algorithm	No	Cellular network and Mobile Edge Computing
Singh et al. [24] (2020)	Not specified	Maximization of the energy efficiency	Mixed-integer non-convex programming, Dinkelbach method, and Iterative Algorithm	No	Multi-user multicarrier amplify-and-forward networks
Jayaweera et al. [67] (2020)	Downlink	Minimization of both maximum decoding error probability and total energy consumption	Symbol Sharing Algorithm	No	Factory floor environment
Nasir et al. [68] (2021)	Downlink	Maximization of the minimum rate among UEs	Path-following Algorithm	No	Wireless communication network
Ren et al. [30] (2020)	Downlink	Minimization of the decoding error probability	Bisection search method, one-dimensional line search method, and Lagrangian dual decomposition method	No	Industrial (Factory)

Table 2.3: Strategies used in wireless communication system for URLLC services.

Proposal	Focus	Objective	Techniques / Algorithms	D2D	Scenario
Librino and Santi [70, 71] (2020, 2021)	Uplink	Allocation of as many devices as possible, considering those that provide the desired reliability with the least amount of resources	Graph-Based Allocation Algorithm (bipartite graph) and Hungarian algorithm	No	Industrial (Factory)
Dong et al. [21] (2021)	Downlink	Minimization of the total power consumption of a BS	Artificial Neural Network (ANN) and MINLP	No	Wireless communication network
Chu et al. [20] (2019)	Not specified	Maximization of the achievable rate of the D2D pair	Difference of convex programming, First-order Taylor approximation, and Iterative successive convex approximation method	Yes	Cellular network
Yang et al. [22] (2020)	Uplink	Minimization of transmission latency and Maximization of the minimum amount of transmission information between all V2V pairs	Binary search and Newton's method, Epigraph Form, Standard geometric programming, and Interior-point methods	Yes	Vehicular Networks
Sanusi et al. [28] (2021)	Uplink	Maximization of the overall system throughput	MINLP and Stable Matching Problem, game theoretical approach	Yes	Industrial (Factory)
I. O. Sanusi, K. M. Nasr, and K. Moessner [29] (2019)	Uplink	Maximization of the overall system throughput	Not specified	Yes	Wireless IoT industrial environment.
Yang et al. [31] (2020)	Uplink and Downlink	Maximization of the energy-efficiency	Deep post-decision state (PDS)-based experience replay and transfer (PDS-ERT) RL algorithm	Yes	Industrial networks
Yang et al. [72] (2019)	Uplink	Maximization of the throughput of admitted D2D links	Reinforcement learning framework	Yes	Internet of Vehicles communication networks
Wu et al. [73] (2020)	Not specified	Maximization of the sum STP of all D2D pairs	Game Theory, sequential spatial adaptive play	Yes	Wireless communication network
Our second RRA proposal	Uplink	Maximization of energy efficiency	Artificial Bee Colony (RRA-URLLC-ABC), Ant Colony System with External Memory (RRA-URLLC-ACS-EM) and Particle Swarm Optimization (RRA-URLLC-PSO)	Yes	Cellular network for an industrial factory scenario

2.3 FINAL CONSIDERATIONS

This section presented the state-of-the-art of the CPP and RRA problems in cellular networks, considering the management of D2D communications. Since the treatment of D2D communications was not focused on solving the SDN CPP problem, we proposed an approach to the problem, which considers two sub-problems. In RRA is solved by a greedy heuristic, obtaining the number of D2D communications accepted in the network. The result was used as input for CPP, which was solved by two meta-heuristics based on ABC and ACS-EM meta-heuristics.

Publications on the RRA problem were reviewed considering different approaches such as D2D communications management and interference, social-aware solutions, and systems for URLLC services. According to the literature review, taking into account the social relationship between users helps the selection of D2D communications that will have assigned resources. Therefore, a social-aware approach has been proposed to solve the RRA problem that considers full and partial CSI models. The partial one includes a feedback model that decreases the network overhead caused by the full CSI model.

In addition, the importance of studying systems for URLLC services and their use with D2D communications were addressed. The literature on URLLC systems was analyzed and the need for studies on algorithms that obtain optimal or near-optimal solutions in considerably low times for not losing the current state of the network was identified. Three bio-inspired ABC-based, ACS-EM-based, and PSO-based meta-heuristics have been proposed for solving the problem.

3 SYSTEM MODELING AND METAHEURISTIC-BASED ALGORITHMS

In this chapter, we present initially a system model and a set of models in order to treat some important parts of the network system. Such set is composed by:

- a social network model;
- a URLLC services model;
- a channel model;
- an interference model;
- a queueing model for SDN controllers; and
- a CSI and Limited selected-*NM* Feedback model.

At the end of the chapter, we present a generic description of the metaheuristics used as a basis for solving the optimization problems discussed in this thesis:

- Artificial Bee Colony (ABC);
- Ant Colony System with External Memory (ACS-EM); and
- Particle Swarm Optimization (PSO).

3.1 SYSTEM MODEL

A cellular network formed by one Packet Data Network Gateway (PGW), a group of Serving Gateways (SGWs) defined as a set $S = \{s_i\} = \{s_1, s_2, \dots, s_s\}$, $1 \leq i \leq s$, a group of eNBs defined as a set $B = \{b_k\} = \{b_1, b_2, \dots, b_b\}$, $1 \leq k \leq b$, and a group of SDN controllers defined as a set $C = \{c_j\} = \{c_1, c_2, \dots, c_c\}$, $1 \leq j \leq c$ was considered. The SDN controllers can be located in any entity in the cellular network (PGW, SGW or eNB). Figure 3.1 shows a simplified representation of our topology for $s = 2$, $b = 4$, and $c = 2$. SGWs are connected to PGW and can communicate via a wired link and eNBs are evenly distributed among the SGWs and can communicate with each other through a wired link.

The system involves multiple D2D pairs and cellular users, as shown in Figure 3.2. User Equipment (UE) is classified as either Cellular User Equipment (CUE), or D2D-capable User Equipment (DUE), according to the type of communication it establishes. If it communicates

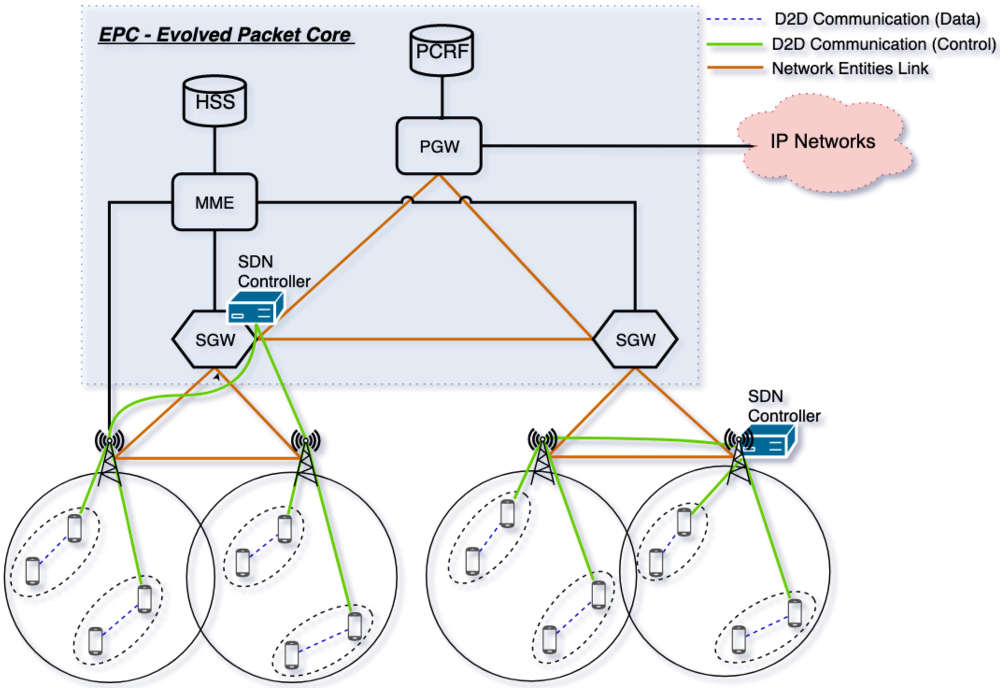


Figure 3.1: SDN Cellular Network for the control of D2D communications.

directly with the eNB, it is classified as CUE, and if it establishes a direct link with another UE, a D2D communication occurs and both UEs are classified as DUEs. Therefore, we define a cellular scenario where cellular and D2D connections coexist in the same cell and transmit over the same bandwidth. The eNB is located in the cell center and the UEs are uniformly random distributed in the cell.

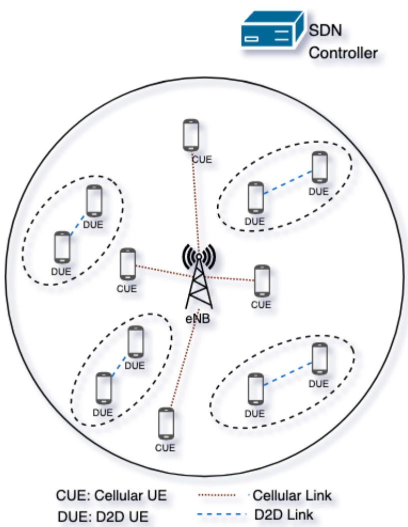


Figure 3.2: D2D and cellular concurrent users inside an SDN Controller domain.

Physical domain assumes a typical Long Term Evolution (LTE) system and the time division duplex (TDD) technique, in which each frame consists of a number of DL, Uplink (UL), and special subframes according to the uplink-to-downlink configuration type. The resources in the

UL subframe are organized into resource blocks (RBs) in the frequency domain, and slots in the time domain [74]. We selected the underlay mode, as in [75, 76, 77, 78, 79], which assumes D2D pairs share the subchannels with CUEs operating in the uplink mode. UL spectrum sharing is considered, since it is under-utilized in comparison to that of DL in cellular systems. Similarly to [80, 81], the network uses the orthogonal frequency-division multiple access (OFDMA) technique to provide communication services, with orthogonal uplink subchannels.

The group of CUEs is defined as a set $U = \{u_i\} = \{u_1, u_2, \ldots, u_u\}$, $1 \leq i \leq u$ and the group of D2D pairs is considered $D = \{d_j\} = \{d_1, d_2, \ldots, d_d\}$, $1 \leq j \leq d$. In a D2D communication, the DUE transmitter is denoted by d_j^T and the receiver is represented by d_j^R. Cellular communications are considered primary and radio resources are pre-assigned to CUEs before sharing a RB with D2D communications. CUEs are assumed to occupy orthogonal channels so that no interference occurs.

Each D2D pair can select one of the following communication modes: (i) RM - the D2D pair shares the channel with either a cellular communication, or another D2D communication; (ii) DM or overlay mode, when the cellular network allocates a fraction of the available resources for the exclusive use of D2D devices (the pair uses a free channel); and (iii) CM - the pair uses the conventional transmission mode by sending data through the eNB (2 uplink channels are allocated). The cellular communication mode is used by the D2D link only if the pair cannot communicate directly, either because the transmitter and receiver are far apart, or because of channel conditions. In this case, as in [72], the eNB is assumed to allocate its resources for guaranteeing the QoS requirements on the downlink.

The bandwidth on UL is divided into orthogonal RBs of equal size; a set $N_{RB} = \{rb_k\} = \{rb_1, rb_2, \ldots, rb_n\}$, $1 \leq k \leq n$, with $n = u + N_u$ RBs is defined, where N_u is the number of unused RBs. When all channels are busy, D2D communications consider the inband underlay mode, as in [78], where D2D pairs reuse the same communication channels - in particular, the same RB - used by CUEs in the licensed frequency bands. Available RBs can be used by D2D communications to complete their transmission.

Let us consider $\psi[u_i, d_j]$ an indicator expressing the reuse of spectral resources, and x_{d_j} indicator denoting the communication mode used by D2D communications, expressed by

$$\psi[u_i, d_j] = \begin{cases} 1, & \textit{if } u_i \textit{ and } d_j \textit{ share the same RB}, \\ 0, & otherwise, \end{cases} \tag{3.1}$$

$$x_{d_j} = \{x_{d_j}^{DM}, x_{d_j}^{RM}, x_{d_j}^{CM}\}, \quad j = 1, 2, \ldots, d, \tag{3.2}$$

where $x_{d_j}^{DM}$, $x_{d_j}^{RM}$ and $x_{d_j}^{CM}$ are equal to 1 when D2D link d_j uses DM, RM or CM, respectively, and equal to 0 otherwise.

Moreover, we assumed all instantaneous channel power gains of CUEs - eNB and d_j^T - eNB links were available at the eNB. In practice, such CSIs can be aware of eNB, since it acts as their receiver, and the CSI of D2D links can be obtained when a D2D connection request is sent. Traditional channel measurement schemes are not applicable to the CSI of CUEs - d_j^R and d_j^T - d_j^R links, since eNB does not participate directly in communication [17]. We assumed only eNB had their location information, which would be used for a partial CSI evaluation.

3.2 SOCIAL NETWORK MODEL

Each user equipment has two roles, i.e., a physical node in the Long Term Evolution Advanced (LTE-A) network, and a user in the virtual social network. The UEs are uniformly distributed in the coverage area of each eNB and can be CUEs or DUEs. Fig. 3.3 shows a two-domain system model composed of physical and social domains. The former considers the processing of SDN controllers and physical links between users. The latter quantifies the social relations between mobile users.

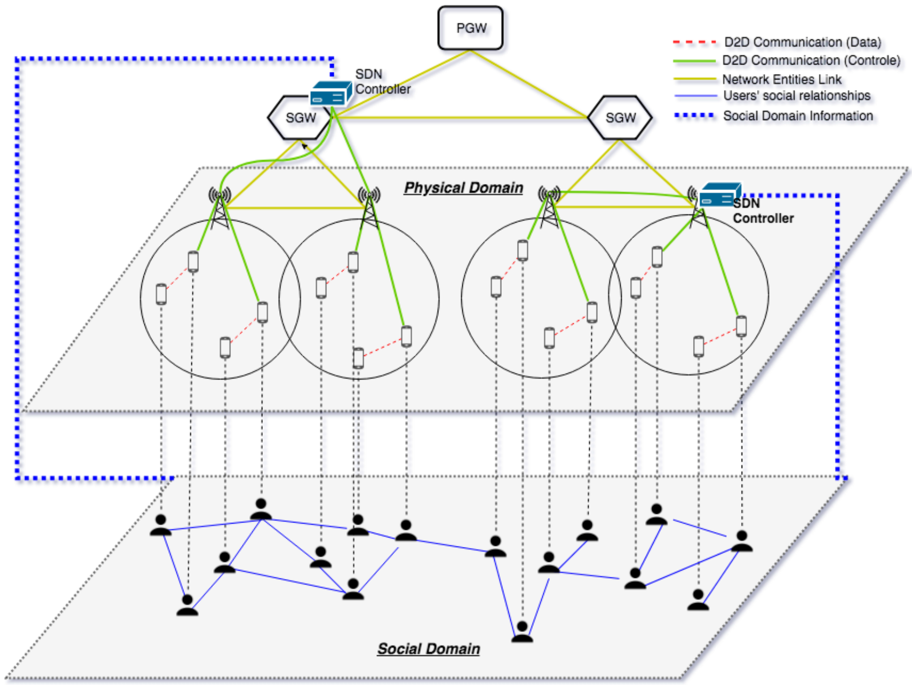


Figure 3.3: SDN Cellular Network for the control of social-aware D2D communications.

Social networks provide relevant information about some patterns of contacts or interactions and common friends, so that the intensity of their social relationships can be determined. We define $\omega_{d_j}^1$ as the common features index and $\omega_{d_j}^2$ as the common friends index to represent the interactivity factor [82]. Features refer to the content type that users access, for example, education, sports, movies, art, among others.

$\omega_{d_j}^1$ measures the strength of the social tie based on the user's common features (content type that both users access) as the ratio between the number of in-common features of a d_j pair users in relation to the total number of features of d_j^T and d_j^R, and it is given by

$$\omega_{d_j}^1 = \frac{|\omega_{d_j^T}^1 \cap \omega_{d_j^R}^1|}{|\omega_{d_j^T}^1 \cup \omega_{d_j^R}^1|} \quad , \tag{3.3}$$

where $|\cdot|$ is the cardinality of set, $\omega_{d_j^T}^1$ and $\omega_{d_j^R}^1$ stand for the features set of d_j^T and d_j^R, respectively.

$\omega_{d_j}^2$ measures the strength of social tie based on friends information as the ratio between the number of in-common friends of d_j users regarding the total number of friends of d_j^T and d_j^R, and it is given by

$$\omega_{d_j}^2 = \frac{|\omega_{d_j^T}^2 \cap \omega_{d_j^R}^2|}{|\omega_{d_j^T}^2 \cup \omega_{d_j^R}^2|} \quad , \tag{3.4}$$

where $\omega_{d_j^T}^2$ and $\omega_{d_j^R}^2$ stand for the friends set of d_j^T and d_j^R, respectively.

Another factor for the determination of degree of relationship between D2D users is reciprocity, measured according to a reciprocity index between users d_j^T and d_j^R, and expressed as

$$\omega_{d_j}^3 = \frac{\omega_{d_j^T \text{ -> } d_j^R}^r + \omega_{d_j^R \text{ -> } d_j^T}^r}{2}, \tag{3.5}$$

where $\omega_{d_j^T \text{ -> } d_j^R}^r = 1$ if user d_j^T follows user d_j^R, or zero, otherwise. The combination of the interactivity factor with the information of reciprocity between two users enables the representation of the strength of social relationships ω_{d_j} as

$$\omega_{d_j} = \frac{1}{3} \cdot \omega_{d_j}^1 + \frac{1}{3} \cdot \omega_{d_j}^2 + \frac{1}{3} \cdot \omega_{d_j}^3, \tag{3.6}$$

where $\omega_{d_j} \in [0, 1]$, and $\omega_{d_j} = 1$ (or $= 0$) denotes the strongest (or weakest) relations between D2D users d_j^T and d_j^R.

3.3 URLLC SERVICES MODEL

D2D communications are assumed to use URLLC services, where users are expected to transmit short packets to ensure low latency requirements. Therefore, the maximum number of bits r_{d_j} carried in a packet of m symbols with decoding error probability ϵ becomes a more accurate performance metric for D2D communications and is given by [25, 27, 19]:

$$r_{d_j} = \log_2(1 + \text{SINR}_{d_j}) - \sqrt{\frac{V}{m}} Q^{-1}(\epsilon) \quad [bps/Hz], \tag{3.7}$$

where $Q^{-1}(\cdot)$ denotes the inverse of *Gaussian Q-function*, m represents the number of transmitted symbols, and V is the channel dispersion expressed as

$$V = \left(1 - \frac{1}{(1 + \text{SINR}_{d_j})^2}\right) \cdot (\log_2 e)^2. \tag{3.8}$$

3.4 CHANNEL MODEL

Channel power gain considers distance-determined path-loss, fast fading due to multi-path propagation, and slow fading due to shadowing. Therefore, such gain between CUE u_i and eNB b_k can be expressed as

$$g_{u_i,b_x} = \beta_{u_i,b_k} \zeta_{u_i,b_k} L_{u_i,b_k}, \tag{3.9}$$

where β_{u_i,b_k} is the fast fading gain with exponential distribution (Rayleigh fading) with unit mean, ζ_{u_i,b_k} is the slow fading gain with log-normal distribution, $L_{u_i,b_k} = 128.1 + 37.6log_{10}(l[Km])$ is the path-loss model for cellular link, and l is the distance between CUE u_i and eNB b_k in Km. Similarly, the channel gain of D2D pair can be expressed by

$$g_{d_j^T,d_j^R} = \beta_{d_j^T,d_j^R} \zeta_{d_j^T,d_j^R} L_{d_j^T,d_j^R}, \tag{3.10a}$$

where $L_{d_j^T,d_j^R}$ is given by

$$L_{d_j^T,d_j^R} = 148 + 40log_{10}(l[Km]). \tag{3.10b}$$

The interference channel gains from d_j^T to eNB b_k, from CUE u_i to d_j^R, and from d' (transmitter of other D2D pair) to d_j^R are denoted by $I_{\triangle u_i,b_k}$, I_{u_i,d_j^R}, and I_{d',d_j^R}, respectively.

3.5 INTERFERENCE MODEL

In the following, first a description of cellular communications is presented, where the interference generated by D2D communications in reuse mode is considered. Next, a description of the model adopted in D2D communications in each communication mode considered is given.

3.5.1 Cellular Communications

A RB used by a CUE can be shared with a D2D pair in reuse mode. This case increases the capacity of the network [83, 84]; however, it also leads to higher interference between entities that use the same RBs for transmission. Therefore, the control of the interference level is essential for an efficient use of D2D in this mode. Based on the interference model described in [85], we considered the interference between cellular and D2D communications in the radio resource allocation process (Figure 3.4).

Therefore, the minimum interference requirements on cellular communication should be considered for the reuse RBs. Consequently, the eNB in radio resource allocation enables the reuse

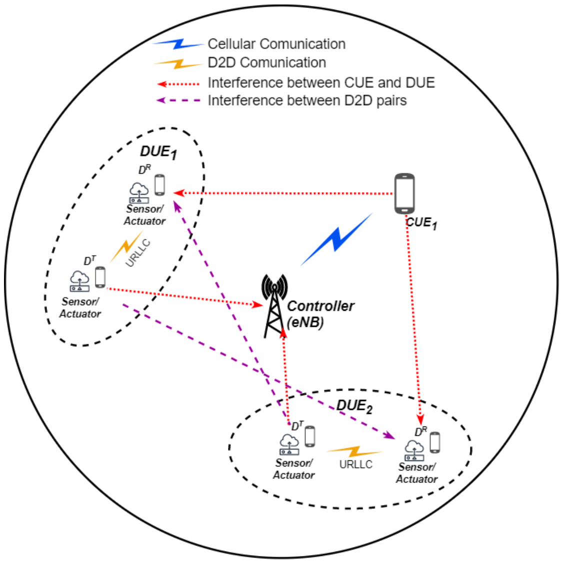


Figure 3.4: Reuse Mode Interference.

of a RB if the SINR requirements of the cellular communication have been satisfied - the SINR_{u_i} received must be greater than the defined SINR threshold, as shown in equation

$$\text{SINR}_{u_i} = \frac{P_{u_i} g_{u_i,b_k}}{I_{\Delta u_i,b_k} + \sigma^2} \geq \text{SINR}_u^t, \tag{3.11}$$

where b_k and u_i denote eNB b_k and CUE u_i, respectively, P_{u_i} is the transmit power of cellular user u_i, g_{u_i,b_x} is the channel gain from transmitter u_i to receiver eNB b_k, $I_{\triangle u_i,b_k}$ is the interference received at eNB b_k from D2D pair transmitters using the RB assigned to u_i, and $\sigma^2 = N_0 W$ [86, 87] is the noise variance assumed the same for eNB and all D2D receivers, N_0 is single-sided noise power spectral density and W is system bandwidth.

Interference ($I_{\triangle u_i,b_k}$) received at the eNB b_k associated with u_i, must be lower than or equal to $I_{u_i}^{max}$ defined by SINR_u^t and given by

$$I_{\Delta u_i,b_k} = \sum_{d_j^T \in \Delta u_i} P_{d_j} g_{d_j^T,b_k} \leq I_{u_i}^{max} \quad , \tag{3.12a}$$

$$I_{u_i}^{max} = \frac{P_{u_i} g_{u_i,b_k}}{\text{SINR}_u^t} - \sigma^2 \quad , \tag{3.12b}$$

where Δu_i is the set of D2D pairs that reuse the RBs allocated for u_i, $g_{d_j^T,b_k}$ is the channel gain from transmitter d_j^T to receiver eNB b_k, and P_{d_j} is the transmitted power of D2D pair d_j.

Furthermore, when an RB used by a CUE is not reused by any D2D pair, it does not suffer interference from D2D pairs. In this case, the Signal-to-Noise Ratio (SNR) received at the eNB for CUE u_i can be given by

$$\text{SNR}_{u_i} = \frac{P_{u_i} g_{u_i,b_k}}{\sigma^2}, \tag{3.13}$$

Based on the interference model, we can derive the maximum number of bits transmitted on cellular links, defined by

$$r_{u_i} = \begin{cases} \log_2(1 + \text{SNR}_{u_i}), & \textit{if } \sum_{j=1}^{d} \psi_{u_i,d_j} = 0, \\ \log_2(1 + \text{SINR}_{u_i}), & otherwise. \end{cases} \tag{3.14}$$

3.5.2 D2D Communications

Three modes of communication for D2D pairs, in which interference is generated differently, were considered. In what follows is a description of the model adopted in each case.

Reuse Mode (RM)

In RM, D2D communications consider the underlay inband mode, as in [78], where D2D links reuse the same communication channels - in particular, the same RB - that use a CUE of the cellular network in licensed frequency bands. The minimum SINR requirements for D2D communications must be considered - a D2D pair d_j can reuse a given RB if $\text{SINR}_{d_j}^{RM}$ is greater than the SINR threshold, defined as

$$\text{SINR}_{d_j}^{RM} = \frac{P_{d_j} g_{d_j^T,d_j^R}}{I_{u_i,d_j^R} + I_{d',d_j^R} + \sigma^2} \geq \text{SINR}_d^t, \tag{3.15}$$

where $g_{d_j^T,d_j^R}$ is the channel gain from transmitter d_j^T to receiver d_j^R; I_{u_i,d_j^R} is the interference received at d_j^R from u_i, and I_{d',d_j^R} is the interference received at d_j^R from D2D pair transmitters using a same RBs as d_j. The above interference can be expressed as

$$I_{u_i,d_j^R} = P_{u_i} g_{u_i,d_j^R} \quad , \tag{3.16a}$$

$$I_{d',d_j^R} = \sum_{d' \in \Delta u_i - d_j^T} P'_d g_{d',d_j^R} \leq I_{d_j}^{max} \quad , \tag{3.16b}$$

$$I_{d_j}^{max} = \frac{P_{d_j} g_{d_j^T,d_j^R}}{\text{SINR}_d^t} - \sigma^2 \quad . \tag{3.16c}$$

Therefore, if two users are in the LTE Direct coverage area, they can initiate a direct link between them whenever the SINR limits have been met. Based on the interference model, we can derive the maximum number of bits transmitted on D2D links when D2D pair is in RM, expressed as

$$r_{d_j}^{RM} = \log_2(1 + \text{SINR}_{d_j}^{RM}) - \sqrt{\frac{V}{m}}.Q^{-1}(\epsilon) \quad [bps/Hz]. \tag{3.17}$$

D2D communications are assumed to use URLLC services and cellular communications use traditional services, which commonly do not require short packets.

Dedicated Mode (DM)

In DM, the D2D pairs use a empty RB. RBs are allocated orthogonally between UEs, and no interference occurs among such UEs. Consequently, the SNR is the metric considered in this mode expressed by

$$\mathrm{SNR}_{d_j}^{DM} = \frac{P_{d_j} g_{d_j^T, d_j^R}}{\sigma^2}, \tag{3.18}$$

where $\mathrm{SNR}_{d_j}^{DM}$ is the SNR of D2D communications in DM. Therefore, the maximum number of bits transmitted by D2D users is given by

$$r_{d_j}^{DM} = \log_2(1 + \mathrm{SNR}_{d_j}^{DM}) - \sqrt{\frac{V}{m}} Q^{-1}(\epsilon). \tag{3.19}$$

Cellular Mode (CM)

In CM, the D2D pair uses the conventional transmission mode by sending data through the eNB (two uplink channels are allocated). The mode can be used by the D2D link if the pair cannot communicate directly, either because the transmitter and receiver are far apart, or because of channel conditions. As in [72], the eNB is assumed to allocate its resources for guaranteeing the QoS requirements on the downlink.

A D2D communication can use the CM if there are at least two unused RBs. Therefore, no interference is generated with other cellular or D2D communications. The metric considered in this mode is SNR, given by

$$\mathrm{SNR}_{d_j^T}^{CM} = \frac{P_{d_j^T} g_{d_j^T, b_k}}{\sigma^2}, \tag{3.20a}$$

$$\mathrm{SNR}_{d_j^R}^{CM} = \frac{P_{d_j^R} g_{d_j^R, b_k}}{\sigma^2}, \tag{3.20b}$$

where $P_{d_j^T}$ and $P_{d_j^R}$ are the transmit power of transmitter and receiver of D2D link d_j, respectively, and $g_{d_j^T, b_k}$ and $g_{d_j^R, b_k}$ are the gain of the channels between eNB and DUEs in CM. Therefore, the maximum number of bits transmitted is given by

$$r_{d_j}^{CM} = r_{d_j^T}^{CM} + r_{d_j^R}^{CM}, \tag{3.21a}$$

$$r_{d_j^T}^{CM} = \log_2(1 + \mathrm{SNR}_{d_j^T}^{CM}) - \sqrt{\frac{V}{m}} Q^{-1}(\epsilon), \tag{3.21b}$$

$$r_{d_j^R}^{CM} = \log_2(1 + \text{SNR}_{d_j^R}^{CM}) - \sqrt{\frac{V}{m}} Q^{-1}(\epsilon). \tag{3.21c}$$

Considering all communications in the network, the system throughput and the total transmission power consumption are expressed as

$$R_T = \sum_{i=1}^{u} r_{u_i} + \sum_{j=1}^{d} x_{d_j}^{RM} r_{d_j}^{RM} + \sum_{j=1}^{d} x_{d_j}^{DM} r_{d_j}^{DM} + \sum_{j=1}^{d} x_{d_j}^{CM} r_{d_j}^{CM} \quad [bps/Hz], \tag{3.22}$$

$$P_T = \sum_{i=1}^{u} P_{u_i} + \sum_{j=1}^{d} x_{d_j}^{RM} P_{d_j} + \sum_{j=1}^{d} x_{d_j}^{DM} P_{d_j} + \sum_{j=1}^{d} x_{d_j}^{CM} (P_{d_j^T} + P_{d_j^R}) \quad [Watt]. \tag{3.23}$$

3.6 QUEUEING MODEL

The SDN controller is treated here as a Markovian model M/ M/m_c/K, with a single queue of limited capacity, where m_c is the number of cores of each controller. Such modeling has been used by other authors such as Gurusamy et al. [88], Panev and Latkoski [89], and Li et al. [90]. We suppose the SDN controllers, used in the cellular network, are running VMs with four Central Process Unit (CPU) cores. Their limit is set by the K parameter and follows a First In First Out (FIFO) discipline.

Let us consider:

- λ_c: average request rate received in the SDN controller;
- λ_{eff}: effective request rate in the SDN controller;
- λ_{lost}: lost requests rate in the SDN controller;
- μ: processing rate of each core, which follows an exponential distribution;
- m_c: number of controller cores ($m_c = 4$); and
- $m_c\mu$: processing rate of the SDN controller.

When the system is full, the new requests are discarded, hence, $\lambda_{eff} = 0$ for $n_q \geq K$, where n_q is the request number received by the controller, and λ_{eff} = λ_c - λ_{lost} and λ_{lost} = $\lambda_c \cdot p_K$, where p_K is the probability of K requests in the system.

The controller utilization rate ρ is given by [91]

$$\rho = \frac{\lambda_{eff}}{m_c \mu} \quad . \tag{3.24}$$

The probability (p_0) of no requests in the controller depending on the utilization rate is given by

$$p_0 = \begin{cases} \left\{ \sum_{n_q=0}^{m_c-1} \frac{(\lambda_c/\mu)^{n_q}}{n_q!} + \frac{(\lambda_c/\mu)^{m_c}}{m_c!} \cdot \left[\frac{1-(\lambda_c/m_c\mu)^{(K-m_c+1)}}{1-(\lambda_c/m_c\mu)} \right] \right\}^{-1}, & \text{if } \frac{\lambda_c}{m_c\mu} \neq 1 \quad , \\ \\ \left\{ \sum_{n_q=0}^{m_c-1} \frac{(\lambda_c/\mu)^{n_q}}{n_q!} + \left[\frac{(\lambda_c/\mu)^{m_c}}{m_c!} \cdot (K - m_c + 1) \right] \right\}^{-1}, & \text{if } \frac{\lambda_c}{m_c\mu} = 1 \quad . \end{cases} \tag{3.25}$$

The average queue length (L_q) is given by

$$L_q = \begin{cases} \frac{p_0(\lambda_c/\mu)^{m_c} \cdot (\lambda_c/m_c\mu)}{m_c! \cdot (1-(\lambda_c/m_c\mu))^2} \cdot \left[1 - \left(\frac{\lambda_c}{m_c\mu}\right)^{K-m_c} - (K - m_c) \cdot \left(\frac{\lambda_c}{m_c\mu}\right)^{K-m_c} \cdot (1 - \frac{\lambda_c}{m_c\mu}) \right], & \text{if } \frac{\lambda_c}{m_c\mu} \neq 1 \quad , \\ \\ \frac{(\lambda_c/\mu)^{m_c}(K-m_c)(K-m_c+1)}{2m_c!} \cdot p_0, & \text{if } \frac{\lambda_c}{m_c\mu} = 1 \quad . \end{cases} \tag{3.26}$$

The probability (p_K) of K requests in the controller as a function of the utilization rate is given by

$$p_K = \frac{(\lambda_c/\mu)^K}{m_c! \cdot m_c{}^{(K-m_c)}} \cdot p_0 \quad . \tag{3.27}$$

Finally, the average response time of each SDN controller (t_{c_j}) is the time spent by the packet in the queue plus the processing time, obtained as

$$t_c = \frac{L_q}{\lambda_{eff}} + \frac{1}{\mu} \quad . \tag{3.28}$$

3.7 CHANNEL STATE INFORMATION AND LIMITED SELECTED-*NM* FEEDBACK MODEL

The Channel State Information (CSI) available at eNB and D2D pairs is different:

At eNB: eNB knows g_{u_i,b_k} and $I_{d_j^T,b_k}$, $d_j^T \in d_j$, $\forall u_i \in U$, $\forall d'^T \in D$. eNB is the receiver in those links and can estimate them using the reference signals transmitted by users.

At D2D Pairs: The d_j^R of D2D pair d_j knows $g_{d_j^T,d_j^R}$, I_{u_i,d_j^R} and $I_{d'^T,d_j^R}$, $\forall d_j \in D, \forall u_i \in U$. It is the receiver in those links and can estimate them using, for example, probing reference signals.

Two approaches, namely full CSI and limited feedback with partial CSI are considered for CSI. The former is applied when an eNB knows the status of all channels on the network, whereas partial CSI considers the sending of limited feedback in bits and number of both CUEs and D2D pairs, so that eNB has partial information from CSI.

In the full CSI approach, the channel sum rate can be achieved if all cellular users and D2D pairs are considered together by Equation 3.22. The feedback overhead can also be accounted as

$$overhead = d * q * [u + (d - 1)], \tag{3.29}$$

where q is the number of bits sent by each D2D pair receiver to eNB containing its quantized CSI information (2^q quantization levels). eNB has the SINR of all links, considering the interference of CUE and D2D pairs sharing a same channel, which is equivalent to setting q to ∞ [16].

The selected-*NM* method is applied to the approach with limited q-feedback, where all D2D pairs use RM and its transmission power is the same. Instead of d_j^R sending information from all CUEs and D2D pairs, it sends information from $N \in U$ cellular users and $M \in D$ D2D pairs. N is the most promising CUE for the reuse of its channel and M denotes the D2D pairs with which it can share the same channel simultaneously. Therefore, the feedback overhead can be accounted as

$$overhead = d * q * [N + M], \tag{3.30}$$

where $N \leq |U|$ and $M \leq |D|$. Up to M D2D pairs are allowed on each channel whenever QoS parameters have been met for both cellular and D2D communications in RM. The channel rate of rb_n is calculated as

$$R_{rb_n}^{RM} = r_{u_i} + \sum_{j=1}^{M} \psi_{u_i,d_j} r_{d_j}^{RM}. \tag{3.31}$$

The key is to find the *N* most potential reuse partner CUEs and *M* more potential shared partner D2D pairs. An intuitive method is to choose the *N* farthest CUEs and the *M* farthest D2D pairs of longest distance from d_j^R. However, in some cases, it may not work, since other factors, such as QoS requirements also affect the interference of D2D pairs.

Inspired in [17, 18], we have derived a minimum distance metric, l^{min}, to evaluate the access ability for a D2D pair. The metric for a $d_j \in D$, D2D pair sharing resources with a CUE $u_i \in U$, l_{u_i,d_j}^{min}, can be expressed as

$$l_{u_i,d_j}^{min} \geq 10^{-(148-X)/40} \quad [Km], \tag{3.32a}$$

$$X = \begin{cases} \frac{P_{u_i} g_{u_i,b_k} g_{d_j^T,d_j^R} - \text{SINR}_u^t \sigma^2 \left(g_{d_j^T,d_j^R} + \text{SINR}_d^t g_{d_j^T,b_k} \right)}{\beta_{u_i,d_j^R} \zeta_{u_i,d_j^R} P_{u_i} \text{SINR}_u^t \text{SINR}_d^t g_{d_j^T,b_k}} & \text{if } \frac{P_{u_i} g_{u_i,b_k}}{P_{d_j} g_{d_j^T,b_k} + \sigma^2} \leq \text{SINR}_u^t, \\ \frac{P_{d_j} g_{u_i,b_k} g_{d_j^T,d_j^R} - \text{SINR}_d^t g_{u_i,b_k} \sigma^2}{\beta_{u_i,d_j^R} \zeta_{u_i,d_j^R} \text{SINR}_u^t \text{SINR}_d^t \left(P_{d_j} g_{d_j^T,b_k} + \sigma^2 \right)} & \text{if } \frac{P_{u_i} g_{u_i,b_k}}{P_{d_j} g_{d_j^T,b_k} + \sigma^2} > \text{SINR}_u^t. \end{cases} \tag{3.32b}$$

The minimum distance metric enables eNB to easily decide whether D2D pair $d_j \in D$ can share the resource with CUE $u_i \in U$. However, the metric depends on fading components β_{u_i,d_j^R}, ζ_{u_i,d_j^R} and the power channel gain of D2D pair $g_{d_j^T,d_j^R}$, which are not known at the eNB. As in [17, 18], we derived a modified distance metric, $\hat{l}_{u_i,d_j}^{min}$, which is the minimum distance between CUE u_i and the receiver of D2D pair d_j that satisfies all QoS parameters without considering (i) the fading effect of u_i on d_j^R link by setting $\beta_{u_i,d_j^R} = 1$, $\zeta_{u_i,d_j^R} = 1$ and (ii) the power channel gain of d_j^T to d_j^R link by setting $g_{d_j^T,d_j^R} = 1$. The modified metric can be expressed as

$$\hat{l}_{u_i,d_j}^{min} \geq 10^{-(148-\hat{X})/40} \quad [Km], \tag{3.33a}$$

$$\hat{X} = \begin{cases} \frac{P_{u_i} g_{u_i,b_k} - \text{SINR}_u^t \sigma^2 \left(1 + \text{SINR}_d^t g_{d_j^T,b_k} \right)}{P_{u_i} \text{SINR}_u^t \text{SINR}_d^t g_{d_j^T,b_k}} & \text{if } \frac{P_{u_i} g_{u_i,b_k}}{P_{d_j} g_{d_j^T,b_k} + \sigma^2} \leq \text{SINR}_u^t, \\ \frac{P_{d_j} g_{u_i,b_k} - \text{SINR}_d^t g_{u_i,b_k} \sigma^2}{\text{SINR}_u^t \text{SINR}_d^t \left(P_{d_j} g_{d_j^T,b_k} + \sigma^2 \right)} & \text{if } \frac{P_{u_i} g_{u_i,b_k}}{P_{d_j} g_{d_j^T,b_k} + \sigma^2} > \text{SINR}_u^t, \end{cases} \tag{3.33b}$$

eNB then selects the N most potential partner CUEs with the N largest $l_{u_i,d_j^R}/\hat{l}_{u_i,d_j}^{min}$ for a D2D pair d_j. The maximum-distance ratio, $l_{u_i,d_j^R}/\hat{l}_{u_i,d_j}^{min}$, was chosen as the user selecting metric for the following reasons:

1. the bigger the ratio, the longer the distance between CUE and the D2D pair receiver, and the more ability to counteract the fading;
2. a lower $\hat{l}_{u_i,d_j}^{min}$ implies CUE can tolerate more D2D interference and obtain higher D2D throughput if CUE is selected.

Similarly, the minimum distance metric for the evaluation of the access ability for a D2D pair $d_{j_1} \in D$ sharing resources with another D2D pair $d_{j_2} \in D$, $l_{d_{j_1},d_{j_2}}^{min}$, can be expressed as

$$l_{d_{j_1},d_{j_2}}^{min} \geq 10^{-(148-Y)/40} \quad [Km], \tag{3.34a}$$

$$Y = min \begin{cases} \frac{P_{d_j} g_{d_{j1}^T,d_{j1}^R} - \text{SINR}_d^t \sigma^2}{\beta_{d_{j_2}^T,d_{j_1}^R} \zeta_{d_{j_2}^T,d_{j_1}^R} P_{d_j} \text{SINR}_d^t}, \\ \frac{P_{d_j} g_{d_{j2}^T,d_{j2}^R} - \text{SINR}_d^t \sigma^2}{\beta_{d_{j_1}^T,d_{j_2}^R} \zeta_{d_{j_1}^T,d_{j_2}^R} P_{d_j} \text{SINR}_d^t}. \end{cases} \tag{3.34b}$$

The QoS parameters must be satisfied for both pairs for the evaluation of multiple D2D pairs that share a same channel; therefore, the most restricted value of Equation 3.34b is considered for the obtaining of Y. The minimum distance metric enables eNB to easily decide whether D2D pair $d_{j_1} \in D$ can share the resource with D2D pair $d_{j_2} \in D$. However, it depends on fading components $\beta_{d^T_{j_2},d^R_{j_1}}$, $\zeta_{d^T_{j_2},d^R_{j_1}}$,$\beta_{d^T_{j_1},d^R_{j_2}}$, $\zeta_{d^T_{j_1},d^R_{j_2}}$, and the power channel gain of D2D pairs, $g_{d^T_{j_1},d^R_{j_1}}$, $g_{d^T_{j_2},d^R_{j_2}}$, which are not known at the eNB. Therefore, a modified distance metric, $\hat{l}^{min}_{d_{j_1},d_{j_2}}$, is derived for satisfying the QoS parameters without considering:

- the fading effect of $d^T_{j_2}$ on $d^R_{j_1}$ link by setting $\beta_{d^T_{j_2},d^R_{j_1}} = 1$ and $\zeta_{d^T_{j_2},d^R_{j_1}} = 1$;
- the fading effect of $d^T_{j_2}$ on $d^R_{j_1}$ link by setting $\beta_{d^T_{j_1},d^R_{j_2}} = 1$ and $\zeta_{d^T_{j_1},d^R_{j_2}} = 1$;
- the power channel gain of $d^T_{j_1}$ on $d^R_{j_1}$ link by setting $g_{d^T_{j_1},d^R} = 1$; and
- the power channel gain of $d^T_{j_2}$ on $d^R_{j_2}$ link by setting $g_{d^T_{j_2},d^R_{j_2}} = 1$.

The modified metric can be expressed as

$$\hat{l}^{min}_{d_{j_1},d_{j_2}} \geq 10^{-(148-\hat{Y})/40} \quad [Km], \tag{3.35a}$$

$$\hat{Y} = \frac{P_{d_j} - \text{SINR}^t_d \sigma^2}{P_{d_j}\text{SINR}^t_d}. \tag{3.35b}$$

eNB then selects the M most potential partner D2D pairs with the M largest $l_{d_{j_1},d_{j_2}}/\hat{l}^{min}_{d_{j_1},d_{j_2}}$ D2D pairs d_{j_1} and d_{j_2}. Similarly to the CUEs selection, maximum-distance ratio, $l_{d_{j_1},d_{j_2}}/\hat{l}^{min}_{d_{j_1},d_{j_2}}$ was chosen and, towards guaranteeing the QoS parameters of the D2D pairs, $l_{d^T_{j_1},d^R_{j_2}}$ must be

$$l_{d_{j_1},d_{j_2}} = min\left(\frac{l_{d^T_{j_2},d^R_{j_1}}}{\hat{l}^{min}_{d_{j_1},d_{j_2}}}, \frac{l_{d^T_{j_1},d^R_{j_2}}}{\hat{l}^{min}_{d_{j_1},d_{j_2}}}\right). \tag{3.36}$$

After the user selection procedure is finished, each D2D pair can trace and report the CSI of the respective N selected CUEs and M selected D2D pairs. When CSI is available at eNB, the radio resource allocation and metaheuristic-based solution can be used. Therefore, d^R_j sends q-bits feedback γ_{d_j} to each selected CUE and D2D pairs to eNB by quantizing SINR_{d_j}. Feedback γ_{d_j} is given by

$$\gamma_{d_j} = i, \quad \text{if } \text{SINR}_i \leq \text{SINR}_{d_j} < \text{SINR}_{i+1}, \tag{3.37}$$

where $0 \leq i \leq 2^q - 1$ and $\text{SINR}_0 = 0 < \text{SINR}_1 < \cdots < \text{SINR}_{2^q-1} < \infty$ are the 2^q quantization thresholds. Given γ_{d_j}, eNB only knows SINR_{d_j} exceeds SINR_{γ_d}. Therefore, using Shannon's formula, eNB assigns the rate of $log_2(1 + \text{SINR}_{\gamma_d})$ to D2D pair d, considering the interference of M D2D pairs selected.

3.8 BIOINSPIRED COMPUTING AND OPTIMIZATION PROBLEMS

Optimization problems consist in finding the best combination of a set of variables to maximize or minimize a function, usually called the objective function. These problems can be divided into three categories: those whose variables assume real (or continuous) values, those whose variables assume discrete (or integer) values and those where there are integer and continuous variables, classified as Continuous Optimization, Combinatorial or Discrete Optimization and Mixed Optimization problems, respectively. In this thesis we study the CPP and RRA problems, which are Combinatorial Optimization problems.

Combinatorial optimization problems can be minimization or maximization problems. In both cases, we have a function applied to a finite domain, which is usually enumerable. Despite being finite, the domain of the function is usually large and algorithms that check each element of this domain become impractical. Thus, the need arises to use more elaborate techniques to find optimal value solutions, which can be of minimum value, if the problem is minimization, or maximum value, if the problem is maximization.

Combinatorial optimization aims to find the best configuration for a project with the most efficiency and the lowest operational cost. Each of the numerous existing optimization methods can achieve better results depending on the type of problem to which they are applied. The choice of method depends on a series of characteristics of the problem to be optimized, mainly the behavior of the objective function. Regarding the relationship between the decision variables in the objective function and the constraints, such methods can be classified into linear programming and non-linear programming methods. Non-linear methods can be divided into deterministic methods and heuristic (stochastic) methods.

Deterministic methods are based on the calculation of derivatives, or approximations thereof, and require information about the gradient vector, either by looking for the point where it cancels out or by using the direction in which it points. These methods produce better results for continuous, convex, and unimodal functions (functions that have only one minimum or maximum point).

Heuristic methods use only the information of the function to be optimized, which can be difficult to represent, non-linear, discontinuous, non-differentiable, multimodal (it has many minimum or maximum points). These methods search for the optimal solution through probability rules operating in a "random oriented" way.

A heuristic aims at obtaining good solutions (approximate solutions), but at an acceptable computational cost (often based on simple and intuitive procedures). In this case, it offers no guarantees of optimality, nor of admissibility, and eventually, it cannot guarantee quality information about a given solution. Therefore, the heuristic method aims to find a feasible solution (within the space of solutions) to a problem with a high degree of complexity - such as, for example, those of the NP-complete or NP-hard class - in a plausible computational time. In order to deepen the search in heuristic methods, making it possible to find better solutions, the technique known as meta-heuristics was originated.

Meta-heuristics use learning tactics that contain mechanisms to avoid confinement to specific parts of the search space in order to identify a solution that is close to the global optimum solution. In this way, the meta-heuristic reduces the search space towards the global optimum, making computational time feasible.

CPP and RRA problems are NP-hard problems, which cannot be solved using traditional deterministic algorithms when the number of network entities increases. Therefore, meta-heuristic methods have been confirmed to be excellent methods for solving these complex optimization problems, and have been applied to solve many such problems belonging to different domains.

A class of meta-heuristics that has received a lot of attention in recent times consists of bio-inspired computing or biology-inspired computing tools. Bio-inspired computing is the line of research that employs metaphors and models of biological systems in the design of computational tools for solving complex problems. Bio-inspired meta-heuristics have been widely used to discover the best solutions to various optimization problems that are not solved by conventional methods. The main advantage of these meta-heuristics is that they perform iterative searches efficiently by utilizing their exploration and exploitation features, i.e., investigating the entire search space and intensifying the search for an optimum in a specific region, respectively.

Bio-inspired meta-heuristics can be widely divided into three groups: evolutionary, swarm intelligence-based, and ecologically inspired [92]. Evolutionary meta-heuristics are inspired by biological evolution in nature. They define a set of optimization techniques that simulate the natural biological evolution and social behavior of living species.

Ecological-inspired meta-heuristic is an assessment to practice the interactions among the species of ecosystem to design and develop cooperative and intelligence algorithms, it depends on populations of individuals and each population develops according to a particular search strategy. In a way, that the individuals of each population are modified according to the mechanisms of diversification, intensification and the initial parameters of the search strategy. The ecological inspiration stems from the use of some ecological concepts, such as: habitats, ecological relationships and ecological successions.

Swarm intelligence is the field of studying and designing well-organized computational intelligent interactive multi-agent systems that cooperate to gather to achieve a specific goal and to solving complex optimizations problems by using the behavior of real living swarms such as birds, fish, bees, and ants. It is a part of Artificial Intelligence introduced by Wang and Beni in 1989 in the global optimization framework as a collection of algorithms for controlling robotic swarms [93].

As swarm intelligence meta-heuristics possess features of self-organization, parallel operations, distributive operations, flexibility, and robustness, they have been gradually very widespread and are utilized in many events. The emergence of these meta-heuristics has provided fast and reliable methods for obtaining solutions to different complex optimization problems. They include

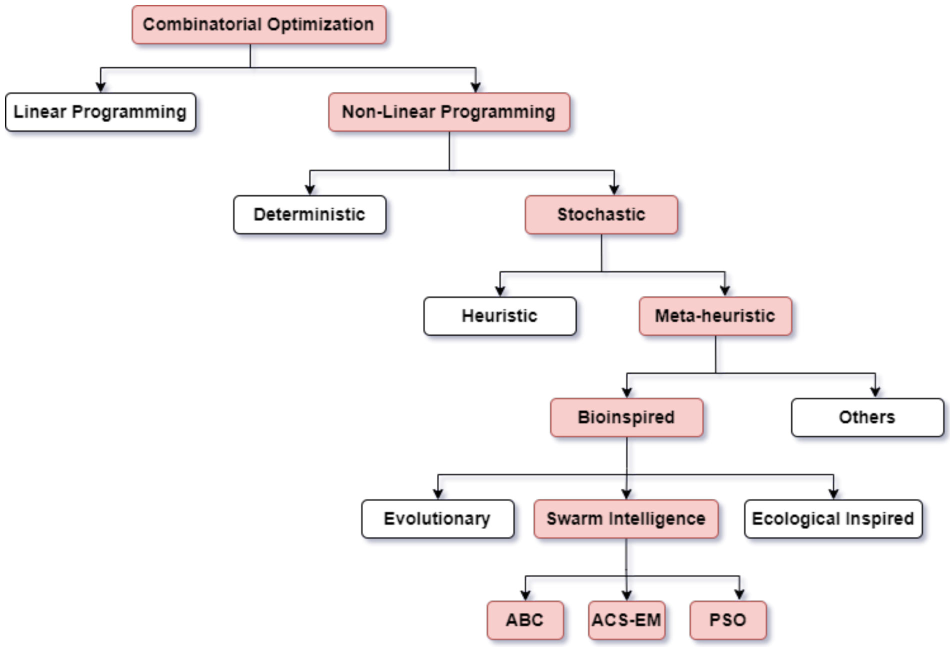


Figure 3.5: Taxonomy of Combinatorial Optimization Methods (based on [95, 92]).

ABC, Ant Colony Optimization (ACO), PSO, Artificial Fish Swarm (AFS), Bacterial Foraging Optimization (BFO), and other meta-heuristics, and reflect their ability to solve non-linear design problems in real-world applications thinking in almost all areas of science, engineering, and industry. These applications include, for example, those related to transportation problems (such as those related to unmanned aerial vehicles), network routing, route planning, robotic systems for scheduling problems, power systems, fault diagnosis, parameter optimization, system identification, cluster analysis, data mining, image processing, layout optimization, and signal processing [94].

According to the literature review conducted by Tang et al. in 2021 in article [94], the swarm intelligence solutions with the highest number of publications until 2020 are PSO, ACO, and ABC, which were the ones used in this study to solve CPP and RRA problems. Figure 3.5 presents a taxonomy of Combinatorial Optimization methods, refining the areas of interest for this study [95, 92].

In this thesis, a study of three well-known meta-heuristic techniques is conducted to examine their suitability for the problems under consideration. This study of meta-heuristic performance is necessary since, according to the no-free-lunch (NFL) theorem [96, 97], there is no a priori superior meta-heuristic technique for dealing with a specific optimization problem.

3.9 ARTIFICIAL BEE COLONY (ABC) ALGORITHM

Karaboga [98, 99] introduced a bee swarm algorithm, known as ABC method, to solve numerical and combinatorial optimization problems. Inspired by the biological behavior of bee colonies in their search for food, it consists of three essential components, namely positions of food sources, quantities of nectar and different types of bees. Each food source position represents a feasible solution to the problem, and the amount of nectar denotes the quality of the solution or fitness. Each bee class represents a particular operation for the generation of new positions of food sources (new solutions).

Each solution within the search space includes a set of parameters that represent the positions of the food sources. The "affinity" value, provided by the objective function, refers to the quality of the food source. In general, the optimization process mimics the search of bees for important food sources, thus, resulting in a process analogous to finding optimal solutions.

The ABC meta-heuristic classifies the foraging artificial bees into three groups, namely, employed bees, onlookers and scouts. A bee currently exploiting a food source is called an employed bee, whereas a bee waiting in the hive for deciding on the choice of a food source is called an onlooker. A bee randomly searching for a new food source is called a scout.

ABC shows a food source associated with each employed bee. Artificial onlooker bees choose sources according to the aptitudes of the solutions (sources) pointed out by the employed bees. The artificial scout bees randomly generate a new candidate solution.

A search requires the exploration and intensification phases to be robust. In ABC, the intensification stage is performed by the employed and onlookers bees, whereas the exploration phase is conducted by the scout bees.

The parameters of the basic ABC meta-heuristic are the number of food sources (SN), which is equal to the number of the employed bees or looker bees, number of trials after which a food source is assumed to be abandoned ($limit$), and a termination criterion. The number of employed bees or onlookers is set equal to the number of food sources in the population.

ABC starts with a population of randomly generated feasible solutions or food sources. Next, the following three steps are repeated until a termination criterion has been met: (i) sending of employed bees to the food sources and measurement of their nectar amounts; (ii) selection of food sources by onlookers after the information from employed bees has been shared and the nectar amount of the food sources has been determined; and (iii) determination of scout bees and their sending to possible food sources.

In Algorithm 1, the ABC workflow (i.e., pseudo-code) with its respective characteristics is represented. The initialization stage of the solutions of random mode is conducted in line 2. The bees phase is, then, carried out: (i) employed bee phase; (ii) onlooker bee phase; and (iii) scout bee phase. Below is the description of each phase.

During the evolution process, an operator from NL is removed and used to generate a new food source for an employed bee or onlooker. If the new food source successfully replaces the current one, the approach enters a winning neighboring list (WNL). Once the NL is empty, it is refilled as follows: 75% are refilled from the WNL list, and the remaining 25% are refilled by a random selection from four different approaches. See [100] for details of the self-adaptive

Algorithm 1 Artificial Bee Colony (ABC)

```
Input: SN, limit, NL_size, nro_it
AllSolutions[SN] ← InitPopulation()
for i = 0 to nro_it do
    for s = 0 to SN do                                              ▷ ***Employed bee phase***
        if OperatorNeighborList == 0 then
            GenerateOperatorNeighborList(NL_size)
        end if
        solution ← AllSolutions[s]
        newSolution ← GetFeasibleNeighbor()
        Fitnes_newSolution ← CalculateOF(newSolution)
        if Fitnes_newSolution ≥ Fitness[solution] then
            AllSolutions[s] ← newSolution
            Fitness[s] ← Fitnes_newSolution
            WNL ← OperatorNeighborList.First
            Limit[solution] ← 0
        else
            Limit[solution] ← Limit[solution] + 1
        end if
        RemoveOperatorNeighborList.First
    end for
    for s = 0 to SN do                                              ▷ ***Onlooker bee phase***
        solution ← ProbabilisticRouletteOperator(Fitness[SN])
        if OperatorNeighborList == 0 then
            GenerateOperatorNeighborList(NL_size)
        end if
        newSolution ← GetFeasibleNeighbor()
        Fitnes_newSolution ← CalculateOF(newSolution)
        if Fitnes_newSolution ≥ Fitness[solution] then
            Solutions[solution] ← newSolution
            Fitness[solution] ← Fitnes_newSolution
            WNL ← OperatorNeighborList.First
            Limit[solution] ← 0
        else
            Limit[solution] ← Limit[solution] + 1
        end if
        RemoveOperatorNeighborList.First

        LocalBestSolution ← GetBestSolution(AllSolutions[SN])
        LocalBestFitness ← GetBestFitness(Fitness[SN])
        if LocalBestFitness > GlobalBestFitness then
            GlobalBestSolution ← LocalBestSolution
            GlobalBestFitness ← LocalBestFitness
        end if
    end for
    higherLimitSolution ← GetHigherLimitSolution()                  ▷ ***Scout bee phase***
    AllSolutions[higherLimitSolution] ← GetRandomFeasibleSolution()
    solution ← AllSolutions[higherLimitSolution]
    Fitness[solution] ← CalculateOF(solution)
    Limit[solution] ← 0
    for s = 0 to SN do
        if Limit[s] ≥ limit then
            solution ← GetRandomFeasibleSolution()
            AllSolutions[s] ← solution
            Fitness[s] ← CalculateOF(solution)
            Limit[s] ← 0
        end if
    end for
end for
Output: GlobalBestSolution
```

strategy.

After the application of the self-adaptive strategy to the current solution, a new solution and a new fitness are generated. The fitness value of the new solution and the fitness value of the old solution are then compared. If the new fitness is better than the old fitness solution, the old one is replaced by the new solution, and the old fitness is replaced by the new fitness. This information is inserted into the population.

If the old solution contains a better fitness value in relation to the new solution, its $limit$ value must be incremented, so that the evolution of that solution is represented.

In onlooker bee phase, in each iteration, an onlooker bee selects a food source using a selection operator based on probabilistic roulette, represented in Equation 3.38. This roulette, measures the fitness quality of each population solution and generates a probabilistic value for each solution.

$$p_i = \frac{Fitness_i}{\sum_{i=1}^{SN} Fitness_i} \tag{3.38}$$

where $Fitness_i$ is the value of the objective function (Equation 4.7) for solution i. After probability values p_i have been generated, a random value is generated within the intervals of p_i values, and tends to move to the position of p_i, which has the best fitness value. However, the choice of the value of highest fitness is not guaranteed. According to this probabilistic choice, a population solution is chosen for the application of the self-adaptive strategy, as used by the employed bee to produce a new neighboring solution. If the new food source obtained is better than or equal to the current one, the new food source replaces the current one and becomes a new member in the population.

In scout bee phase, the $limit_i$ values are analyzed and the $limit$ that has reached the threshold value of a non-evolving solution attempts will be chosen. The $limit_i$ solution is abandoned and a new solution is generated and allocated in its place in the population of bees. As in the basic ABC algorithm, at most one food source is generated by a scout bee in each iteration.

3.10 ANT COLONY ALGORITHM WITH EXTERNAL MEMORY (ACS-EM)

In optimization meta-heuristics based on ant colonies (Ant Colony Optimization (ACO)) [101], artificial ants build a solution to a combinatorial problem by walking through a graph called "construction graph", $G_C(V_{tx}, \Lambda)$ that consists of a set of vertices V_{tx} and a set of arcs Λ. Ants move from one vertex to another along the arc of the graph, incrementally constructing a partial solution, and deposit a certain amount of pheromone $\Delta\tau$ in the components, i.e, in the arches they cross. The amount depends on the quality of the solution found. Ants use pheromone information as a guide to travel through the most promising regions of the search space.

ACO meta-heuristics generate candidate solutions for an optimization problem by a construction mechanism, by which the choice of a solution component to be added at each stage. The following rule is used in the intensification of solutions: by construction it is probabilistically influenced by pheromone traces and heuristic information [101]. This study analyzes the possibility of alternating the way solution components are chosen, introducing an external memory as an auxiliary mechanism for making decisions at each stage of the construction of a solution.

The memory structures used allow ants, on certain occasions, not to make decisions randomly, but to make a choice of components of solutions deterministically, influenced by the values recorded in the said memory. Because this memory stores specific search history information from the beginning of the algorithm, it allows to effectively focus on regions of the unvisited search space or, on the contrary, focus on regions already visited and promising. These uses of memory reflect the mechanisms of intensification and diversification.

A frequency-based memory [102], which stores components of the solutions most frequently chosen, was used. It "prohibits" an ant from choosing a solution component, because it is often chosen in the solutions. It is sought through the "prohibition" for generating solutions that effectively differ from those already generated, thus, expanding the exploration of the search space. On the other hand, this information can be used to "promote" a choice because it is considered attractive. Most ants choose it as part of their solutions and, therefore, it should be considered part of a new solution.

During the construction of a solution, ants apply a choice of the action rule similar to that used in the Ant Colony System (ACS) [101]. A parameter $q_0 \in [0, 1]$ is introduced and promotes the choice of a solution component that favors intensification or diversification in the search for solutions. When an ant chooses a component from a solution, it first generates a uniformly distributed random number in the $[0, 1]$ range. If the number is lower than the value of the q_0 parameter, a component is chosen to perform a local search. If the generated number is greater than the q_0 parameter, the component is chosen favoring the search for new solutions. Each of such options includes a decision on traces of pheromones or external memory. Therefore, a vertex v_{txj} is chosen from a vertex v_{txi} according to the following rule

$$v_{txj} = \begin{cases} R_I, & \text{if } rnd_q < q_0 \text{ (Intensification)}, \\ R_D, & \text{if } rnd_q \geq q_0 \text{ (Diversification)}. \end{cases} \tag{3.39}$$

With a fixed probability q_0, the ant chooses the "best possible element" according to the acquired knowledge. It can be based on either external memory, or traces of pheromone, whereas with probability $(1 - q_0)$, it performs a controlled exploration of new solutions, where rnd_q is a uniformly distributed random variable in the $[0, 1]$ range.

The following rule is used in the intensification of solutions:

$$R_I = \begin{cases} arg\ \text{máx}_{j \in N_i^k}\{memory\_frec[j]\}, & \text{if } rnd_r < r_0, \\ arg\ \text{máx}_{j \in N_i^k}\{\tau_{ij} \cdot \eta_{ij}^{\beta}\}, & \text{if } rnd_r \geq r_0, \end{cases} \tag{3.40}$$

where N_i^k is the neighborhood reachable by ant i when it lies in node v_{txi}, τ_{ij} is the amount of pheromone between nodes v_{txi} and v_{txj}, η_{ij} is the heuristic or desirability information between nodes v_{txi} and v_{txj} and β is a parameter that defines the relative importance of heuristic information. A fixed probability r_0 $(0 \leq r_0 \leq 1)$ chooses the element most often selected from the current location, whereas with probability $(1 - r_0)$, the most desirable element is chosen according to the pheromone stroke. rnd_r is a uniformly distributed random variable in the $[0, 1]$ range.

The following rule is used in the exploration of new solutions (diversification):

$$R_D = \begin{cases} \textit{arg}\ \text{mín}_{j \in N_i^k} \{memory\_frec[j]\}, & \text{if } rnd_z < z_0, \\ Z, & \text{if } rnd_z \geq z_0. \end{cases} \tag{3.41}$$

A fixed probability z_0 $(0 \leq z_0 \leq 1)$ enables the selection of the least frequently chosen element from the current location, whereas probability $(1 - z_0)$ promotes the choice of element Z according to the basic selection rule, as in the ACS [101]. rnd_z is a uniformly distributed random variable in the $[0, 1]$ range. Z is selected by the roulette method, according to the probability distribution p_z similarly to the ant system meta-heuristic [101]:

$$p_z(i, j) = \begin{cases} \frac{\tau_{ij}^{\alpha} \cdot \eta_{ij}^{\beta}}{\sum\limits_{j' \in N_i^k} \tau_{ij'}^{\alpha} \cdot \eta_{ij'}^{\beta}}, & \text{if } j \in N_i^k, \\ 0, & \text{otherwise}, \end{cases} \tag{3.42}$$

where α is the parameter that establishes the relative importance of pheromone tracks. The pheromone tracks are updated in two steps: (i) each time an ant traverses an arc, called a local update, and (ii) at the end of each iteration, called global update.

The online or local pheromone update rule is applied by all ants whenever they cross an arc (v_{txi}, v_{txj}) during the solution construction, calculated as

$$\tau_{ij} = (1 - \delta) \cdot \tau_{ij} + \delta \cdot \tau_0, \tag{3.43}$$

where τ_0 is the initial pheromone value and $\delta \in (0, 1)$ is the pheromone evaporation coefficient.

The step-by-step update rule includes both pheromone evaporation and pheromone deposition. Since the pheromone deposited is very small, the application of this rule causes traces of pheromones between the paths traveled by ants to decrease, which leads to an additional exploration technique of the algorithm. The paths traversed by a large number of ants become less attractive to the other ants that cross them in the current iteration. Consequently, they may not follow the same path.

In the global update of pheromone tracks, only the path taken by the ant with the best solution (the *best-so-far* ant) is updated after each iteration by

$$\tau_{ij} = \begin{cases} (1-\delta)\tau_{ij} + \delta\Delta\tau_{ij}^{bs}, & \text{if} \quad j \in \Omega, \\ (1-\delta)\tau_{ij}, & \text{if} \quad j \notin \Omega, \end{cases} \tag{3.44}$$

where $\Omega = \{1, 2, \ldots, J\}$ is the set of nodes that belong to the path traveled by the *best-so-far* ant and $\Delta\tau_{ij}^{bs}$ is the increment in the pheromone tracks expressed as:

$$\Delta\tau_{ij}^{bs} = \frac{1}{OF^{bs}}, \tag{3.45}$$

where OF^{bs} is the value of the objective function of the best iteration solution.

The pseudo-code of the Ant Colony System with External Memory (ACS-EM) is shown in Algorithm 2.

3.11 PARTICLE SWARM OPTIMIZATION (PSO)

PSO is a bio-inspired meta-heuristic based on animal families such as birds and fishes, which move in complex trajectories in a dynamic, organized, and coordinated way, only understanding and knowing the position and speed of their companions. The algorithm, proposed by Kennedy and Eberhart [103], mimics birds social behavior, optimizing surviving advantages when solving the problem of finding a safe point to land.

The goal of the classical PSO is to find a variable vector $P = \{p_j\} = [p_1, p_2, \ldots, p_p]$, $1 \leq j \leq p$ that minimizes or maximizes an objective or fitness function $OF(P)$. P represents birds landing positions and $OF(P)$ assesses how good or bad a landing position $P = \{p_j\} = [p_1, p_2, \ldots, p_p]$ is.

Given a swarm of f particles, with $1 \leq i \leq f$, vector $P^i = \{p_j^i\} = \left(p_1^i, p_2^i, \cdots, p_p^i\right)$, and vector $\Upsilon^i = \{\upsilon_j^i\} = \left(\upsilon_1^i, \upsilon_2^i, \cdots, \upsilon_p^i\right)$ denotes the position and velocity of particle i and $p = u + d$ represents the number of variables to be determined in the RRA-URLLC problem. Set P represents the RBs assigned (or not) to cellular and D2D links. Therefore, each position accounts for a problem solution.

At each iteration, velocity and position are updated according to equations 3.46a and 3.46b.

$$\Upsilon_j^i = \omega\Upsilon_j^i + c1 * rnd_1(pbest_j^i - P_j^i) + c2 * rnd_2(gbest_j - P_j^i), \tag{3.46a}$$

$$P_j^{i+1} = P_j^i + \Upsilon_j^i. \tag{3.46b}$$

The first term of Equation 3.46a is a product between parameter ω (inertia) and particle's previous velocity. If $\omega = 1$, the particle goes in the same direction of the previous motion,

Algorithm 2 Ant Colony System with External Memory (ACS-EM)

Input: $ant\_quantity, \alpha, \beta, q_0, \tau_0, \sigma, nro\_it, G_C(V_{tx}, \Lambda)$

$\tau[\cdot] \leftarrow InitPheromone(\tau_0)$
$memory[\cdot] \leftarrow InitMemory()$
for $i = 0$ to nro_it **do**
  for each $a \in ant\_quantity$ **do**
    $currentPosition \leftarrow RootNode$
    $TourList \leftarrow currentPosition$
    while $currentPosition.Children \neq 0$ **do**
      $\eta[\cdot] \leftarrow CalculateChildrenHeuristic()$
      $rnd_q \leftarrow RandomDouble()$
      if $rnd_q < q_0$ **then**
        $rnd_r \leftarrow RandomDouble()$
        if $rnd_r < r_0$ **then**
          $nextPosition \leftarrow arg\_\max\{memory[i, :]\}$
        else
          $nextPosition \leftarrow arg\_\max\{\tau_{ij} * \eta_{ij}^{\beta}\}$
        end if
      else
        $rnd_z \leftarrow RandomDouble()$
        if $rnd_z < z_0$ **then**
          $nextPosition \leftarrow arg\_\min\{memory[i, :]\}$
        else
          $p_z[\cdot] \leftarrow CalculateChildrenProbability(\alpha, \beta, \tau, \sigma)$
          $nextPosition \leftarrow Get\_Z(p_z)$
        end if
      end if
      $TourList.Add(nextPosition)$
      $SelectRandomPowerValue(nextPosition)$
      $SelectCommunicationMode(nextPosition)$
      $ValidateFeasibleSolution(TourList, n)$
      $\tau_{ij} = (1 - \delta) \cdot \tau_{ij} + \delta \cdot \tau_0$
      $currentPosition \leftarrow nextPosition$
    end while
    $TourList\_OF \leftarrow CalculateOF(TourList)$
    if $TourList\_OF > BestTour\_OF$ **then**
      $BestTour \leftarrow TourList$
      $BetsTour_OF \leftarrow TourList\_OF$
    end if
  end for
  $\tau_{ij} = \begin{cases} (1-\delta)\tau_{ij} + \delta\Delta\tau_{ij}^{bs}, & \text{if} \quad j \in \Omega \\ (1-\delta)\tau_{ij}, & \text{if} \quad j \notin \Omega \end{cases}$

  $memory[\cdot]$ = UpdateMemory(BestTour)
  if $BestTour\_OF > GlobalBestTour\_OF$ **then**
    $GlobalBestTour \leftarrow BestTour$
    $GlobalBestTour\_OF \leftarrow BestTour\_OF$
  end if
end for

Output: $GlobalBestTour$

whereas if $0 \leq \omega < 1$, it goes to other regions in the search domain. Therefore, the inertia parameter controls the exploratory characteristics of the swarm.

The second term considers the difference between the i particle's best position ($pbest_j^i$) and its current position (P_j^i) for attracting the particle to its best known position. $c1$ is a positive constant defined as an individual-cognition parameter to represent the importance of the particle's own previous experiences. rnd_1 is a random value between $[0, 1]$ that avoids premature convergences.

The third term represents social learning so that the particles share information on the best point achieved ($gbest_j$). Similarly to the second term, it attracts the particles to the best solution found until actual iteration. $c2$ behaves as a social learning parameter and rnd_2 plays the same role as rnd_1.

As addressed elsewhere, CPP and RRA are discrete optimization problems; therefore, a discretized PSO was considered. Following the proposal of Chen et al. [104], positions and velocities are represented as crisp sets and the set with possibilities, respectively.

Equation 3.47 defines the set with possibilities A_p for crisp set E, where each element $e \in E$ has a possibility $p(e) \in [0, 1]$ in A_p.

$$A_p = \{e/p(e)|e \in E\}. \tag{3.47}$$

The approach also considers equations 3.46a and 3.46b, but redefining the terms' operators as described in what follows. The multiplication of a coefficient and velocity turns into a multiplication of a coefficient and a set with possibilities, as defined in equations 3.48a and 3.48b.

$$a_c\, A_p = \{e/p'(e)|e \in E\}, \tag{3.48a}$$

$$p'(e) = \begin{cases} 1, & a_c\, p(e) > 1, \\ a_c\, p(e), & otherwise, \end{cases} \tag{3.48b}$$

where $a_c \geq 0$ is a given coefficient.

The difference between two positions turns into the substraction of two crisp sets A_{p_1} and A_{p_2}, as defined in equation 3.49.

$$A_{p_1} - A_{p_2} = \{e|e \in A_{p_1}, e \notin A_{p_2}\}. \tag{3.49}$$

Equations 3.50a and 3.50b redefine the multiplication of a coefficient and a position (crisp set).

$$a_c E' = \{e/p'(e)|e \in E\}, \tag{3.50a}$$

$$p'(e) = \begin{cases} 1, & e \in E', a_c > 1, \\ a_c, & e \in E', a_c \in [0,1], \\ 0, & otherwise. \end{cases} \tag{3.50b}$$

Finally, the addition of two velocities turns into an addition of two sets of possibilities $\Upsilon_1 = \{e/p_1(e)|e \in E\}$, and $\Upsilon_2 = \{e/p_2(e)|e \in E\}$, as defined by equation

$$\Upsilon_1 + \Upsilon_2 = \{e/max(p_1(e), p_2(e)|e \in E)\}. \tag{3.51}$$

The pseudo-code of the PSO is shown in Algorithm 3.

3.12 FINAL CONSIDERATIONS

This chapter described the system model and the suite of network and communication models important in the study of the CPP and RRA optimization problems, to be presented in the following chapters. 6 (six) specific models were discussed: a social network model, a URLLC services model, a channel model, an interference model, a queueing model for SDN controllers, and a CSI and Limited selected-*NM* Feedback model.

Additionally, 3 (three) bio-inspired meta-heuristics that solved the optimization problems under study were described, with their respective pseudocodes.

Algorithm 3 Particle Swarm Optimization (PSO)

Input: $particle\_quantity, \omega, c1, c2, nro\_it$

```
InitVelocity()
InitPosition()
for k = 0 to nro_it do
    for each i ∈ particle_quantity do

***Velocity Updating***
        rnd_1 ← RandomValue()
        rnd_2 ← RandomValue()
        pbest ← Substract(BestSoFarPosition, currentPosition)
        gbest ← Substract(GlobalBestPosition, currentPosition)
        for each p_j ∈ P do
            υ_1 ← Multiply(c1 * rnd_1, pbest[p_j])
            υ_2 ← Multiply(c2 * rnd_2, gbest[p_j])
            υ_3 ← Multiply(ω, currentPosition[p_j])
            υ^i_j ← Sum(υ_1, υ_2, υ_3)
        end for

***Position Updating***
        rnd_3 ← RandomNumber()
        for each υ^i_j ∈ Υ^i do
            cut(υ^i_j) ← {e|e_j p(e) ∈ Υ^i_j and p(e) ≥ rnd_3}
        end for
        new_P^i ← ∅
        for each υ^i_j ∈ Υ^i do
            Cand_Set^i_j ← {e|e ∈ cut(Υ^i_j)) and e satisfies constrains}
            while construction of new_P^i and Cand_Set^i_j ≠ ∅ do
                new_P^i ← SelectRandomElement(Cand_Set^i_j)
                Update(Cand_Set^i_j)
            end while
            if construction of new_P^i is not finished then
                Cand_Set^i_j ← {e|e ∈ P^i_j and e satisfies constrains}
                while construction of new_P^i and Cand_Set^i_j ≠ ∅ do
                    new_P^i ← SelectElement(Cand_Set^i_j)
                    Update(Cand_Set^i_j)
                end while
            end if
            if construction of new_P^i is not finished then
                Cand_Set^i_j ← {e|e ∈ E_j and e satisfies constrains}
                new_P^i ← SelectBestElement(Cand_Set^i_j)
            end if
        end for
        ValidateFeasibleSolution(new_P^i, n)
        P^i ← new_P^i
        BestSoFar_OF ← CalculateOF(P^i)
        if BestSoFar_OF > GlobalBestPosition_OF then
            GlobalBestPosition ← BestSoFar
            GlobalBestPosition_OF ← BestSoFar_OF
        end if
    end for
```

Output: $GlobalBestPosition$

4 CONTROLLER PLACEMENT PROBLEM

This chapter addresses the formulation and solution of the Controller Placement Problem (CPP) for (i) determining the number of controllers, (ii) determining the locations of controllers, and (iii) performing the eNB-controller assignment. The problem is treated as a discrete optimization problem which is solved by ABC and ACS-EM meta-heuristics.

4.1 OPTIMIZATION PROBLEM

An optimization problem has the form ($f(x)$, Π, Φ), where

- $f(x)$ is the objective function that must be minimized or maximized along with the decision variables;
- Π is the input to the problem that defines the search space; and
- Φ is a set of constraints that involves the decision variables which, in turn, affects the size of the search space.

Objective Function - $f(x)$

This work aims to find the minimum number of required Software Defined Networking (SDN) controllers, their optimal location and assignment to eNBs. The optimality criterion is based on the satisfaction of a response time lower than or equal to a specific QoS time, at a lost requests rate in the SDN controller lower than a value $\lambda_{lost}^{threshold}$. $t_{r_{c_j}}$ was defined as the average request response time of a SDN controller c_j and t_{QoS} was the QoS time considered.

$t_{r_{c_j}}$ depends on the average response time of the SDN controller (t_{c_j}) and propagation latency in packet delivery, as follows:

$$t_{r_{c_j}} = t_{c_j} + t_{b_k c_j}. \tag{4.1}$$

The propagation latency $t_{b_k c_j}$, between the eNB b_k and the SDNcontroller c_j, is obtained as the sum of the number of hops between controller location and eNB, multiplied by the weight associated with each link on the path (without loss of generality, all links were considered equal).

After the average request response time ($t_{r_{c_j}}$) has been calculated, the cost function that checks whether the time constraint has been satisfied must be defined and should grow rapidly when the response time starts to be higher than t_{QoS}. Therefore, an exponential function was applied to characterize this behavior, similarly to the approach adopted in Farshin and Sharifian [1]; when the time was higher than t_{QoS}, a constant ξ was used and increased the cost function at a high rate. The cost function (Figure 4.1) can be calculated as

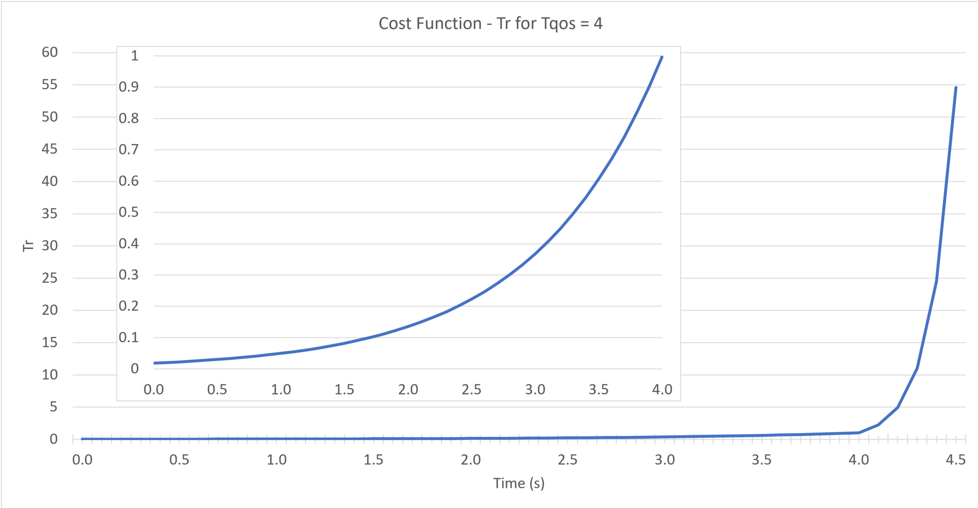


Figure 4.1: Example of average request response time function.

$$T_r(t_{r_{c_j}}) = \begin{cases} e^{(t_{r_{c_j}} - t_{QoS})}, & t_{r_{c_j}} \leq t_{QoS}, \\ e^{\xi(t_{r_{c_j}} - t_{QoS})}, & t_{r_{c_j}} > t_{QoS}, \end{cases} \tag{4.2}$$

where ξ denotes a constant factor that specifies the increase/decrease rate of the cost function, defined as $\xi = 2t_{QoS}$. The total average request response time (T_r) is defined as the average request response time of each controller as

$$T_r = \frac{1}{c}\sum_{j=1}^{c} T_r(t_{r_{c_j}}); \quad c = \text{number of controllers, and } c \geq 1. \tag{4.3}$$

In the controller location problem, the metric most commonly used is the latency between the switches (eNBs) and the controller [34]. It is defined as the average propagation latency between the location of the controller and the location of the eNBs assigned to it [32, 43, 105, 106, 45, 44, 34], and can be obtained as

$$\pi^{avglat}(\Theta) = \frac{1}{b}\sum_{k=1}^{b} \underset{\theta_i \in \Theta}{min} \quad t_{b_k \theta_i}, \tag{4.4}$$

where the set $\Theta = \{\theta_i\} = \{\theta_1, \theta_2, \ldots, \theta_\theta\}$, $1 \leq i \leq \theta$, ($\Theta = PGW \cup S \cup B$), was defined as the possible locations of the controllers, i.e., the controller can be located in any network entity and $t_{b_k \theta_i}$ is the propagation latency between eNB b_k and the controller located in θ_i. The objective is to minimize the average propagation latency given by

$$\pi_C^{avglat} = \frac{1}{c}\sum_{j=1}^{c}\left(\frac{1}{b}\sum_{k=1}^{b}\left(\underset{\theta_{i,j} \in \Theta}{min} \quad t_{b_k \theta_{i,j}}\right)\right), \tag{4.5}$$

such that a given controller cannot be placed in multiple locations and each eNB will be managed only by a controller c_j.

Another important factor in the assignment of eNBs to SDN controllers is the fair distribution of eNBs among controllers, which results in load balancing among them. Jain's Fairness Index (j_f) [107] measures fairness in the allocation of controllers and guarantees that condition. We assumed the average load of the controllers could be calculated as $\rho_c = \frac{\lambda_c}{m_c \mu}$ [108], and the Jain's index could be obtained by

$$j_f = \frac{\left(\sum_{j=1}^{c} \rho_{c_j} \right)^2}{c \cdot \sum_{j=1}^{c} \rho_{c_j}^2}. \tag{4.6}$$

Since our interest was to minimize the lack of fairness among SDN controllers, we consider $J_f = 1 - j_f$ a factor to be minimized. It must be noted that the fairness objective involves a search for a balanced distribution of loads among the set of controllers, and does not involve, necessarily, the load balance of the quantity of D2D pairs attended in each cell.

A function that provides the number of controllers used is required; it must increase, as the number of controllers increases, and be between $[0, 1]$. Therefore, we defined $C_c = 1 - 1/c$, $c \geq 1$, where c is the number of controllers.

Finally, the objective function (Eq. 4.7) considered the weighted sum of the number of controllers, the normalized response time (Eq. 4.3), the normalized average propagation latency between the controllers and the assigned eNBs (Eq. 4.5), and justice parameter (J_f), such that

$$OF = f(x) = \gamma_c \cdot C_c + \gamma_T \cdot T_r + \gamma_\pi \cdot \pi_C^{avglat} + \gamma_F \cdot J_f, \tag{4.7}$$

where γ_c, γ_T, γ_π, and γ_F are the weights of each cost function.

The Controller Placement Problem can be modelled as

$$Minimize \quad (\gamma_c \cdot C_c + \gamma_T \cdot T_r + \gamma_\pi \cdot \pi_C^{avglat} + \gamma_F \cdot J_f). \tag{4.8}$$

Decision Variables

Decision variables describe the quantities that the decision makers would like to determine. Finding the optimal values of the decision variables is the goal of solving an optimization model. In the CPP optimization model addressed in this research, decision variables are:

- cardinality of set C - c;
- the assignment of eNBs and SDN controllers - $M(b, c)$; and
- the SDN controller location - $M(c, \theta)$.

The assignment of eNBs and SDN controllers is denoted by the b x c matrix $M(b,c)$. Binary value of $M(b_k,c_j)$ denotes whether eNB b_k is assigned to SDN controller c_j, where $M(b_k,c_j)$ $= 1$ means yes and 0 otherwise.

The location of SDN controllers is denoted by the c x θ matrix $M(c,\theta)$. Binary value of $M(c_j,\theta_i)$ denotes whether SDN controller c_j is located in network entity θ_i, where $M(c_j,\theta_i) = 1$ means yes and 0 otherwise.

Input Parameters - Π

In order to formulate the mathematical model, we assume the following information is known:

- network topology (number and location of all network entities (eNB, SGW, and PGW), the location of UEs);
- the average control requests rate generated by each D2D pair - λ;
- the number of D2D pairs allowed in each eNB - C_{D2D};
- queueing parameters;
- the maximum response time allowed in the network - t_{QoS}; and
- the maximum lost request rate allowed for the SDN controller - $\lambda_{lost}^{threshold}$.

Constraints - Φ

- Guarantee that the average request response time of SDN controller is lower than or equal to a specific QoS time

$$t_{r_{c_j}} \leq t_{QoS}. \tag{4.9}$$

- Ensure that the lost requests rate in the SDN controller is lower than a threshold value

$$\lambda_{lost} \leq \lambda_{lost}^{threshold}. \tag{4.10}$$

- Guarantees that each eNB will be managed only by a SDN controller

$$\sum_{j=1}^{c} M(b_k,c_j) = 1 \qquad \forall b_k \in B. \tag{4.11}$$

- Guarantees that a given controller cannot be placed in multiple locations

$$\sum_{i=1}^{\theta} M(c_j,\theta_i) = 1 \qquad \forall c_j \in C. \tag{4.12}$$

- Numerical constraints

$$M(b_k,c_j) \in \{0,1\}, \quad M(c_j,\theta_i) \in \{0,1\} \quad \forall b_k \in B, \forall c_j \in C, \forall \theta_i \in \Theta. \tag{4.13}$$

4.2 OVERVIEW OF THE PROPOSED SOLUTION

To solve the CPP we present a solution that integrates radio resource allocation and D2D communications control. Our approach considers two steps as shown in Figure 4.2:

- 1st step: the network topology considering an LTE-A network, the physical layer parameters and a conflict graph (to represent the interference between cellular and D2D communications), were defined; two greedy heuristics are used for radio resources allocation to obtain the set of D2D pairs admitted by eNB; and
- 2nd step: the network topology, considering an LTE-A network, the queue parameters (related to the modeling of the SDN controllers), and the set of D2D pairs admitted to the network, were defined; two bio-inspired meta-heuristics are used to solve the CPP and we obtain the number of SDN controllers, the eNB-controller assignment and the location of the controllers on a cellular network.

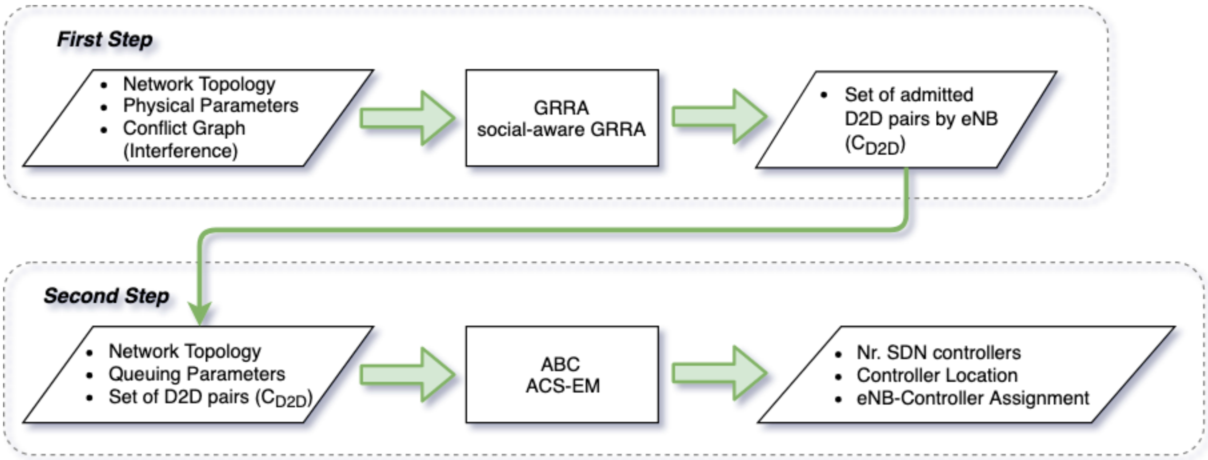


Figure 4.2: Steps involved in CPP solution.

4.3 FIRST STEP: DETERMINATION OF C_{D2D}

In the first step, the number C_{D2D} of simultaneous D2D communications that can be established in each cell is calculated. To obtain C_{D2D}, the resource allocation problem is formulated to maximize the number of admitted D2D pairs in the network, subject to the constraints of the maximum tolerable interference level for both eNB and DUE.

When D2D links are added to the network, two main levels of interference can be found. One interference is caused by the cellular network (i.e. from CUEs to D2D receivers) and the other one is caused by D2D network (i.e from D2D transmitter to the eNBs and from D2D transmitter to D2D receivers of other links). Therefore, the constraints can be summarized as follows:

- $\text{SINR}_{u_i} \geq \text{SINR}_u^t \quad \forall u_i \in U$ (Equation 3.11);
- $\text{SINR}_{d_j} \geq \text{SINR}_d^t \quad \forall d_j \in D$ (Equation 3.15) and
- each D2D pair can only reuse one CUE's resource block.

The serving eNB is assumed to obtain the channel state information for all links and to know the SINR threshold of all user equipment. The purpose is to allow multiple D2D pairs to reuse the same RBs that have been pre-assigned to CUEs, whenever QoS requirements are met. To face this issue, we used a conflict graph [109] to model the network interference.

Let $U = \{u_i\} = \{u_1, u_2, \ldots, u_u\}$ denotes the CUEs, and l_{u_i,b_k} denote the distance between CUE u_i and evolved NodeB b_k. In the eNB area, a cellular link established between CUE u_i and eNB is defined as lk_{C_i}, where $1 \leq i \leq u$, as show in Figure 4.3. A D2D link is defined as $lk_{D_j} = d_j$, where $1 \leq j \leq d$. In Conflict Graph (CG), each link lk is taken as a vertex, and two vertices are connected with an edge if the corresponding links cannot be used for simultaneous transmission.

In the wireless network, CG can be constructed among links sharing the same resources, e.g., RBs in the LTE-A. For a better understanding in Figure 4.3, we show on the left the existing interference in an LTE-A network and on the right the generated CG from it. In CG there is an edge between the vertices lk_{D1} and lk_{D2} because they may not be active simultaneously. An edge is draw if any of the following is true: $l_{d^T_{j2},d^R_{j1}} \leq IR_{d^R_{j1}}$ or $l_{d^T_{j1},d^R_{j2}} \leq IR_{d^R_{j2}}$, where l denotes the distance between transmitter and receiver and IR denotes the interference range. In the example shown in Figure 4.3, $l_{d^T_{j1},d^R_{j2}} \leq IR_{d^R_{j2}}$ for this, an edge exists between the vertices lk_{D1} and lk_{D2}. The model encompasses the case where a conflict arises because links lk_{D1} and lk_{D2} have a UE in common (i.e., $d^T_{j1} == d^T_{j2}$ or $d^T_{j1} == d^R_{j2}$ or $d^R_{j1} == d^T_{j2}$ or $d^R_{j1} == d^R_{j2}$).

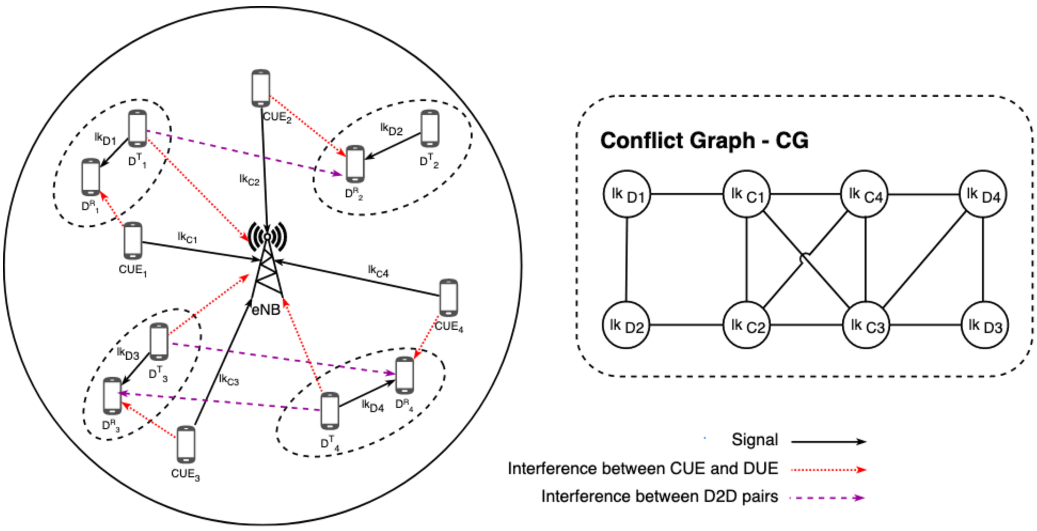


Figure 4.3: Example of the general conflict graph built from an LTE-A network.

In our proposal, a conflict graph is constructed for each CUE, where it is considered the set of D2D links that can share the same resources, that is, RBs. In the CG, each vertex indicates a D2D communication link, and each edge represents the unacceptable interference between the vertices to which it connects [77, 110]. Thus, in each conflict graph, the vertices represent the D2D links that do not conflict with the CUE that uses this RB, and the edges represent the mutual interference between the D2D links. Figure 4.4 shows an example of the CG of 4 CUEs. In CU$E_1 - CG$ the D2D links lk_{D2}, lk_{D3}, and lk_{D4} can share a same RB as CUE u_1. However, links lk_{D3} and lk_{D4} cannot share a same RB. Therefore, a RB assigned to CUE u_1 can be used at the same time by links lk_{D2} and lk_{D3} or lk_{D2} and lk_{D4}.

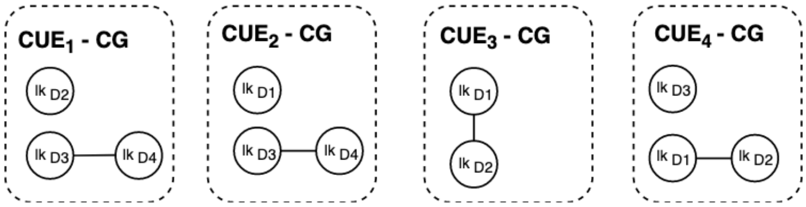


Figure 4.4: Example of the CUEs conflict graph built from an LTE-A network.

To solve the problem presented in the first part of our proposal, we used two greedy heuristics. One was based on the Greedy Resource Allocation Algorithm (GRAA), proposed in [75]. The other was an update of this heuristic considering the social relationship between users, called social-aware GRAA.

In the heuristics, a sequential resource allocation mode was considered, where eNB determines the admitted D2D pairs according to the RB index. Conflict graphs are updated whenever a D2D pair is selected to use a resource block and tolerable interference values are evaluated by both eNB and DUE that have already been admitted. The above procedures are implemented sequentially in each RB and the D2D pairs.

In GRAA, D2D pairs are selected based on the degree of interference. Then, the D2D pairs with the least degree of interference are chosen first. This greedy solution maximizes the total admitted D2D pairs, but does not consider the system throughput. Therefore, we realized a variant of the heuristic that allows achieving a content delivery with high user satisfaction, employing social-aware D2D techniques and, at the same time, maximizing the transmission rate of D2D links, named SA-GRAA.

In the SA-GRAA, D2D pairs are selected based on the intensity of the social relationships (ω_{d_j}) between DUEs. Therefore, the D2D pairs with the highest intensity of social interaction are selected first. The problem involves the social layer and the physical layer, where the transmission rate is formulated as a weighted channel rate, i.e., the rate weighted by the intensity of the social relationship, as in [111]. For each D2D pairs, the transmission rate (r_{d_j}) is obtained by Shannon's theorem as

$$r_{d_j} = W_{RB} \log_2(1 + \text{SINR}_{d_j}), \tag{4.14}$$

where W_{RB} is the system bandwidth per RB, and the weighted throughput ($R^w_{d_j}$) of the link between DUE transmitter d_j^T and DUE receiver d_j^R when reusing resource block rb_k can be obtained by [111]

$$\begin{aligned} R_{d_j}^w &= \omega_{d_j}\ r_{d_j} \\ &= \omega_{d_j}\ W_{RB} \log_2 \left(1 + \frac{P_{d_j}\ g_{d_j^T, d_j^R}}{I_{u_i, d_j^R} + I_{d', d_j^R} + \sigma^2}\right). \end{aligned} \tag{4.15}$$

The meta-heuristics runs on each eNB and the set of admitted D2D pairs in each eNB is returned.

The obtained C_{D2D} values are used as input parameters in the second step of the proposed solution. They are used to define the average control requests rate received by SDN controllers as

$$\lambda_c = \sum_{k=1}^{b_{c_i}} C_{D2D_k} * \lambda_k. \tag{4.16}$$

where λ_k is the average control requests rate generated by each D2D pair ($\lambda_k = \lambda$, in the evaluation), b_{c_i} is the number of eNBs managed by controller c_i and C_{D2D_k} is the number of D2D communications established in eNB b_k. The set of requests sent to a controller involves types such as D2D service registration setup, resource allocation, offloading, connection release, among others. Moreover, C_{D2D_k} will depend on the geographical location of the users and the radio resources allocation on the LTE-A network.

4.4 SECOND STEP: CPP SOLUTION

In the second step, the CPP problem is solved aiming to minimize the response time in the control of D2D communications, while attending the mentioned constraints in Section 4.1. This time considers the average response time of each SDN controller obtained using the queuing model described in the previous section (Equation 3.28). Therefore, the set of pairs D2D admitted in each cell is an input parameter of the second step and included in the response time.

Some optimization problems have been considered "NP-hard" (as channel assignments, cellular network planning), and the use of bio-inspired meta-heuristics has been considered for providing optimal or near-optimal solutions, in an acceptable computation time. Considering the modeling previously presented, the problem to be solved can be formulated as discussed in Section 4.1:

Given: network topology, λ, C_{D2D}, queueing parameters, t_{QoS} and $\lambda_{lost}^{threshold}$

Problem: $Minimize \quad (\gamma_c \cdot C_c + \gamma_T \cdot T_r + \gamma_\pi \cdot \pi_C^{avglat} + \gamma_F \cdot J_f)$

Subject to:

$$C1: \quad t_{r_{c_j}} \leq t_{QoS} \quad \forall c_j \in C;$$

$$C2: \quad \lambda_{lost} \leq \lambda_{lost}^{threshold};$$

$$C3: \quad \sum_{j=1}^{c} M(b_k, c_j) = 1; \quad \forall b_k \in B;$$

$$C4: \quad \sum_{i=1}^{\theta} M(c_j, \theta_i) = 1; \quad \forall c_j \in C;$$

$$C5: \quad M(b_k, c_j) \in \{0, 1\}; \quad \forall b_k \in B, \forall c_j \in C;$$

$$C6: \quad M(c_j, \theta_i) \in \{0, 1\}; \quad \forall c_j \in C, \forall \theta_i \in \Theta.$$

Output: c; $M(b, c)$; $M(c, \theta)$

Two meta-heuristics have been implemented for the solution of CPP:

- the CPP-ABC meta-heuristic and
- the CPP-ACS-EM meta-heuristic.

4.4.1 CPP Artificial Bee Colony (CPP-ABC) Algorithm

The modeling of the solution to the problem under study is defined in ABC. The solution to the CPP problem would be a list with the sequences of the eNB-controller assignments ($b_k c_j$) followed by the location of each controller ($c_j \theta_i$). In Figure 4.5, the modeling adopted is illustrated and it also represents a possible solution used to solve the CPP problem under study. The sequence of nodes modeled in Figure 4.5 represents in ABC a food source or a possible solution to the problem. At the end of each iteration the controllers that did not have assigned eNBs are eliminated from the solution.

b_1c_1	b_2c_4	b_3c_6	b_4c_1	b_5c_3	b_6c_5	b_7c_8	b_8c_1	$c_1\theta_2$	$c_3\theta_4$	$c_4\theta_8$	$c_5\theta_6$	$c_6\theta_3$	$c_7\theta_7$

Figure 4.5: Modeling the solution for the CPP problem.

In the employed bee phase, each solution is updated by a neighborhood structure compose by operators of permutation. In this operation with a fixed probability n_0 ($0 \leq n_0 \leq 1$) a new eNB-controller assignment is performed; while with probability $(1 - n_0)$, the location of a controller is changed. The partial clone and random operators are defined by performing a new assignment of eNB b_i. The insert, swap, and random operators are defined by placing a controller in a different position in the network.

In the partial clone operator, two positions are randomly chosen. After the two positions of the solution vector have been defined, the content of the first position defined (relative to a controller) is copied for the content of the second position. Figure 4.6 shows an example of the partial clone operator applied to the eNB-controller assignment.

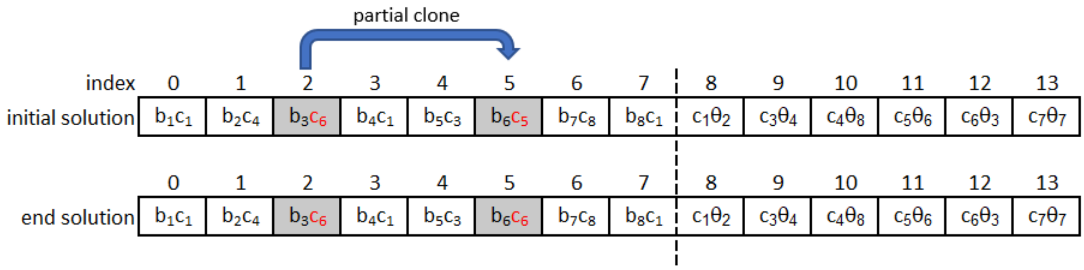


Figure 4.6: Example of a partial clone neighborhood structure.

On the other hand, a position in the solution vector is randomly chosen in the random operator, and the eNB of that position is assigned to another randomly chosen SDN controller, as shown in Figure 4.7.

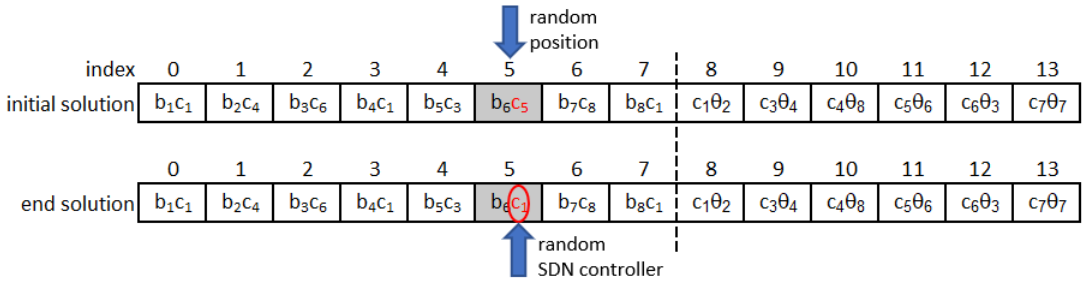


Figure 4.7: Example of a random neighborhood structure.

The insert operator is defined by placing a controller in a different position in the network. In the insert method two positions are chosen randomly. In Figure 4.8, indices 9 and 12 are chosen for this example. In this method the content of the smaller index is moved to the position of the (larger index + 1), in other words, the content of the smaller index is inserted in the position next to that of the larger index. With the application of the insert method, in Figure 4.8 it is possible to observe in the final solution the aforementioned displacement and with a darker color the nodes that had to be shifted so that the insert rule was performed in the correct way .

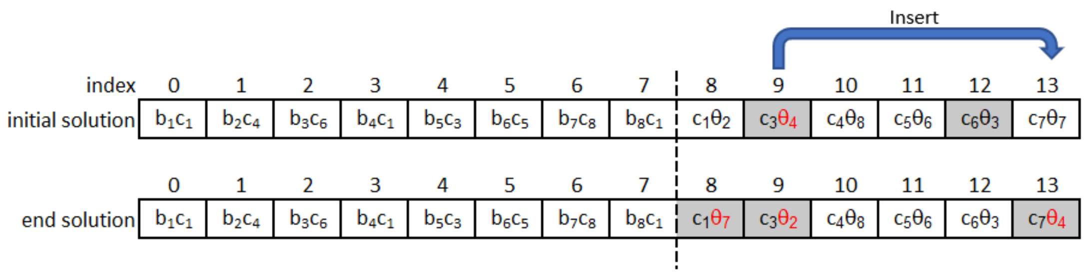


Figure 4.8: Application example of the INSERT neighborhood structure.

In the swap method two positions are chosen randomly analogous to the insert method. In this method with the two positions of the vector defined, the next action is to exchange the content of the first position defined with the content of the second position. To facilitate the understanding of this process in Figure 4.9 there is an example of the swap method applied to the CPP problem.

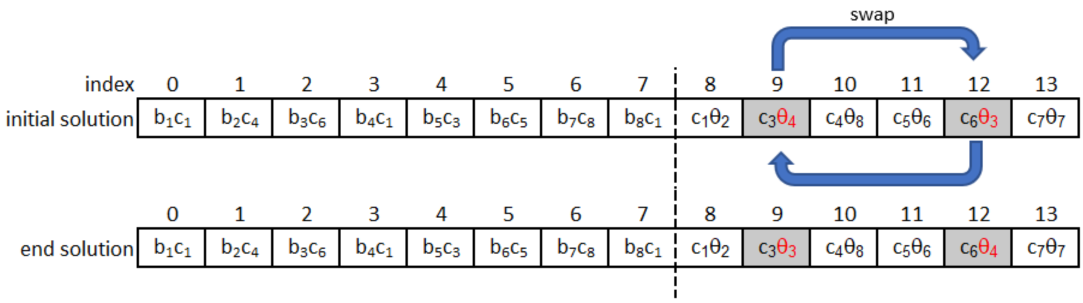


Figure 4.9: Application example of the SWAP neighborhood structure.

Both employed and onlookers bees apply a self-adaptive strategy to find nearby food sources. At first, an initial operator neighbor list (NL) of a specified length (NL_{size}) is generated by randomly populating the list with the neighboring selection approaches, according to the part of the solution considers, as follows:

- **for eNB-controller assignment:**
  1. performing a partial clone operator for a solution;
  2. performing a random operator for a solution;
  3. performing two partial clone operators for a solution;
  4. performing two random operators for a solution.
- **for SDN controller location:**
  1. performing a insert operator for a solution;
  2. performing a swap operator for a solution;
  3. performing a random operator for a solution;
  4. performing two insert operators for a solution;
  5. performing two swap operators for a solution;
  6. performing two random operators for a solution.

After the application of the self-adaptive strategy to the current solution, a new solution and a new fitness are generated. The fitness value of the new solution and the fitness value of the old solution are then compared. If the new fitness is better than the old fitness solution, the old one is replaced by the new solution, and the old fitness is replaced by the new fitness. This information is inserted into the population.

If the old solution contains a better fitness value in relation to the new solution, its $limit$ value must be incremented, so that the evolution of that solution is represented.

4.4.2 CPP Ant Colony System with External Memory (CPP-ACS-EM) Algorithm

In this study, the construction graph was divided into two parts for troubleshooting the CPP. The first part assigns eNBs to controllers and vertex subset $V_{b_k,c_j} = \{b_1c_1, \cdots, b_1c_c, b_bc_1, \cdots, b_bc_c\}$ represents all possible pairs of eNB-controller that exist in the scenario considered. In the second part, controllers are located in the cellular infrastructure. Vertex subset $V_{c_j,\theta_i} = \{c_1\theta_1, \cdots, c_1\theta_\theta, c_c\theta_1, \cdots, c_c\theta_\theta\}$ represents all possible controller-location pairs, as shown in Figure 4.10.

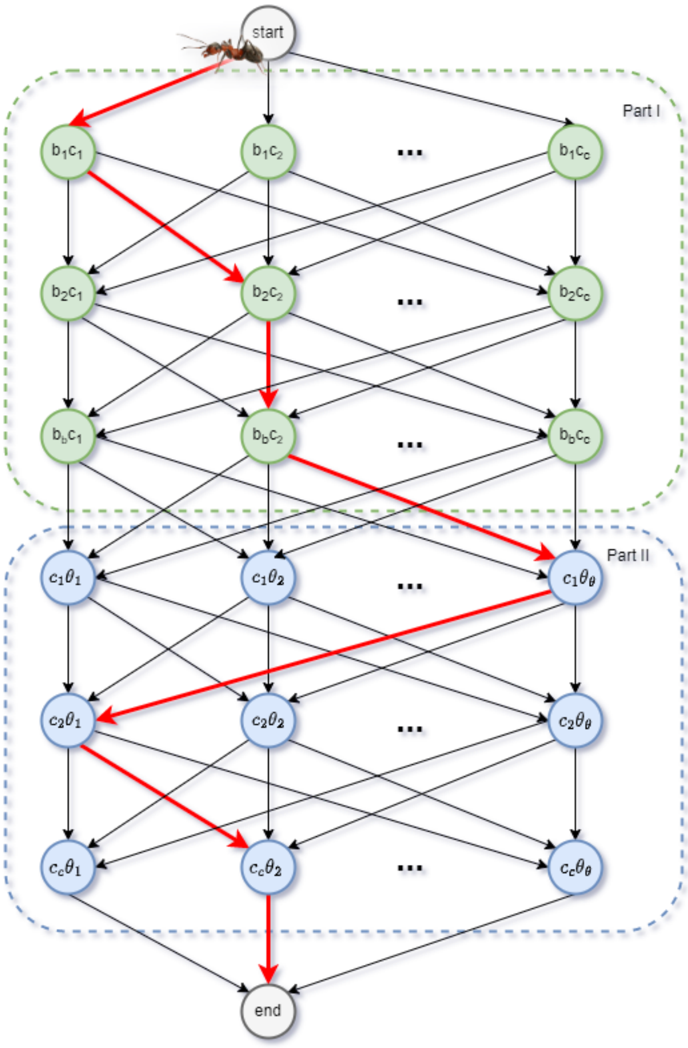


Figure 4.10: Construction graph to CPP.

To obtain the number of SDN controllers used, the graph is constructed assuming an SDN controller is necessary for managing the traffic of each eNB ($c = b$). When the ants reach the end of the path, SDN controllers that have not assigned eNB are eliminated from the solution, and the number of controllers actually used is obtained.

Frequency-based memory stores the number of times the eNB b_k was assigned or the c_j controller. So, this memory allows ants to make decisions taking into account the eNBs that are most often assigned to a given SDN controller, because they have associated a high frequency of assignment; or, conversely, to make decisions taking into account the eNBs are less often assigned to a given SDN controller, because they associated a low frequency of assignment.

The meta-heuristic has a frequency array of size $[b, c]$, called $memory\_frec$, associated. Position $memory\_frec[b_k, c_j]$ stores the number of times eNB b_k was assigned to controller c_j during the execution. It represents the vertices chosen by ants in the construction of the solution and is used in the construction of solutions in two possible ways, i.e., intensification, by the choice of a node that matches as an eNB-controller which was most often selected from the current location, and diversification, by the choice of a node less frequently chosen from the current location. External memory values are updated at the end of each iteration considering the value of the best solution.

The following rule is used in the intensification of solutions:

$$R_I = \begin{cases} \textit{arg}\ \text{máx}_{j \in N_i^k}\{memory\_frec[b_k, c_j]\}, & \text{if } rnd_r < r_0, \\ \textit{arg}\ \text{máx}_{j \in N_i^k}\{\tau_{ij} \cdot \eta_{ij}^{\beta}\}, & \text{if } rnd_r \geq r_0. \end{cases} \tag{4.17}$$

The following rule is used in the exploration of new solutions (diversification):

$$R_D = \begin{cases} \textit{arg}\ \text{mín}_{j \in N_i^k}\{memory\_frec[b_k, c_j]\}, & \text{if } rnd_z < z_0, \\ Z, & \text{if } rnd_z \geq z_0. \end{cases} \tag{4.18}$$

The η_{c_j} heuristic is obtained from the controller utilization rate, ρ_{c_j} (equation 3.24), considering the eNB assignments already performed on the components of the path traveled. Thus, controllers with a lower value of ρ will have a greater desirability, such as the equation:

$$\eta_{c_j} = \begin{cases} 1 - \rho_{c_j}, & \text{if } \rho_{c_j} < 1, \\ 0, & \text{otherwise.} \end{cases} \tag{4.19}$$

4.5 FINAL CONSIDERATIONS

This chapter presented the formulation of the CPP problem as an optimization problem that minimizes an objective function, which considers the weighted sum of the number of controllers, response time, average propagation latency between the controllers and the assigned eNBs, and justice parameter. The result is: the number of SDN controllers needed, the assignment of eNBs and SDN controllers, and the location of SDN controllers.

The proposed solution approach for the CPP problem is also described. It includes two steps, with the result of the first step being one of the inputs needed in the second step. The first step solves the RRA problem and determines the number of D2D communications accepted in the network, with this it is possible to specify the average control requests rate received by SDN controllers. The second step solves the CPP problem aiming to minimize the response time in the control of D2D communications.

Finally, the modeling of the meta-heuristics for the CPP problem is described. The meta-heuristics CPP-ABC and CPP-ACS-EM were used to solve the problem.

5 RADIO RESOURCE ALLOCATION PROBLEM

This chapter presents a study of the radio resource allocation problem modeled as an optimization problem according to two approaches. In the first, the objective is to obtain the number of communications admitted in the network that maximizes the system throughput. The solution involves physical and social layers, considering partial CSI and the limited selected-NM feedback model described in Section 3.7 in the physical layer. The second approach aims at obtaining the number of communications D2D admitted in the network that maximizes its energy efficiency (η_{EE}). URLLC services are used in D2D communications and power control and radio resource allocation are performed for both cellular and D2D communications. Two meta-heuristics based on ABC, called SA-RRA-ABC (based on Social Awareness) and RRA-URLLC-ABC solved the optimization problems. Moreover, the RRA-URLLC-ACS-EM and RRA-URLLC-PSO meta-heuristics were used in the second proposed approach.

5.1 FIRST RRA OPTIMIZATION PROBLEM: MAXIMIZING THE WEIGHTED SYSTEM THROUGHPUT

The goal is to obtain the number of D2D communications admitted in an LTE-A network that maximizes the system throughput at high QoS. The resource allocation problem was formulated towards maximizing the weighted system throughput in the network, subject to the constraints of the maximum tolerable interference level for eNB and DUE.

The optimization problem considered involved both social and physical layers and the objective function was formulated as a weighted system throughput (R_T^w) based on the weighted channel rate. The rate was defined in [111] as the channel rate (r_{d_j}) weighted by the intensity of the social relationship between DUE d_j^T and DUE d_j^R, expressed by

$$R_{d_j}^w = \omega_{d_j}\ r_{d_j}; \quad [bit/(s*Hz)]. \tag{5.1}$$

Therefore, the weighted throughput of D2D link d_j reusing rb_k was obtained by

$$R_{d_j}^w = \omega_{d_j} W_{rb_k} \log_2\left(1 + \frac{P_{d_j} g_{d_j^T, d_j^R}}{I_{u_i, d_j^R} + I_{d', d_j^R} + \sigma^2}\right); \quad [bit/s], \tag{5.2}$$

where W_{rb_k} is the system bandwidth per RB. Similarly, the throughput of the cellular link between cellular user u_i and the respective eNB is given by

$$R_{u_i} = W_{rb_k} r_{u_i} = W_{rb_k} \log_2\left(1 + \frac{P_{u_i} g_{u_i, b_k}}{I_{\triangle u_i, b_k} + \sigma^2}\right); \quad [bit/s]. \tag{5.3}$$

Both throughput of cellular links and weighted throughput of D2D links were considered for the maximization of the weighted system throughput. The optimization problem was formulated as

$$
\begin{aligned}
& \max \quad R_T^w = \sum_{i=1}^{u} \left(R_{u_i} + \sum_{j=1}^{d} \psi[u_i, d_j] \; R_{d_j}^w \right) \\
\text{s.t. } & \textit{C1: } \text{SINR}_{u_i} \geq \text{SINR}_u^t \quad \forall u_i \in U; \\
& \textit{C2: } \text{SINR}_{d_j} \geq \text{SINR}_d^t \quad \forall d_j \in D; \\
& \textit{C3: } \sum_{i=1}^{u} \psi[u_i, d_j] = 1, \quad \forall d_j \in D; \\
& \textit{C4: } \psi[u_i, d_j] \in \{0, 1\}, \quad \forall d_j \in D, \quad \forall u_i \in U; \\
& \textit{C5: } \omega_{d_j} \in [0, 1], \quad \forall d_j \in D.
\end{aligned}
\tag{5.4}
$$

Constraints *C1* and *C2* guarantee the QoS requirements of cellular links and D2D links, respectively. *C3* ensures each D2D pair can reuse only one CUE's resource block. *C4* and *C5* are numerical constraints.

An ABC approach, called SA-RRA-ABC, solved the problem. A sequential resource allocation mode was considered and eNB determined the admitted D2D pairs according to the RB index. Conflict graphs are updated whenever a D2D pair has been selected to use a RB and tolerable interference values are evaluated by both eNB and other DUEs admitted. SA-RRA-ABC runs on each eNB and the set of admitted D2D links is returned.

5.1.1 Social-aware RRA Artificial Bee Colony (SA-RRA-ABC) Algorithm

The modeling of the solution to the problem under study is firstly defined in ABC. The solution to the social-aware RRA problem would be an array with the RB used by D2D links, where index represents all existing D2D links and the value is the index of RB used by the D2D link. -1 indicates no RB has been assigned to the link, therefore, no D2D communication can occur. Figure 5.1 shows the model adopted, which represents a possible solution to the RRA problem under study, and the sequence of nodes modeled denotes a food source in ABC, or a possible solution.

d_1	d_2	d_3	d_4	d_5	d_6	d_7	d_8	d_9	d_{10}
rb_1	rb_4	rb_1	rb_2	-1	rb_3	rb_4	rb_1	rb_3	rb_1

Figure 5.1: Modeling the solution for the social-aware RRA problem.

At the beginning, SN solutions are randomly generated according to the function shown in Algorithm 4 (used in Algorithm 1 - line 2), in which a RB is randomly selected for each D2D pair (Algorithm 4 - lines 3-5) and follows a uniform distribution. The set $N'_{RB} = N_{RB} \cup \{-1\} = \{-1, rb_1, rb_2, \ldots, rb_n\}$, where $n = u$ and $\{-1\}$ denotes no RB has been assigned to the D2D communication. The meta-heuristic checks if the generated solution is feasible according to the constraints defined in Equation 5.4. The SINR values are then checked for both cellular and D2D pairs allocated for a same RB in the solution (Algorithm 4 - lines 6-10).

When the feedback model is considered, a solution is feasible if eNB has received the CSI feedback from the D2D communication receiver, considering the interference from the CUE and other D2D pairs allocated in the same RB. Otherwise, the solution is not feasible and a new one must be generated. eNB makes a request for CSI to the receiver of D2D communication, considering the distance metrics established in subsection 3.7 (Equations 3.33 and 3.35) to define parameters N and M of the feedback model.

Algorithm 4 - InitPopulation()

1: Input: SN
2: **for** $s = 0$ to SN **do**
3:     **for each** $dlink \in D2DLinks$ **do**
4:         $foodSource[dlink] = Select\_Random\_RB(N'_{RB})$
5:     **end for**
6:     **for** $i = 0$ to $foodSource$ **do**
7:         $dlink \leftarrow i$
8:         $rb \leftarrow foodSource[i]$
9:         $Check\_SINR\_Constraints(dlink, rb)$
10:     **end for**
11:     $Solution[s].Add(foodSource)$
12: **end for**
Output: $Solution[\cdot]$ =0

In the bee phase employed, each solution is updated by a neighborhood structure composed of operators of permutation. Clone, insert, swap, and random operators are defined through the selection of a RB to be reused by each D2D link. After an operator has found a neighboring solution, the new solution is validated if it is a feasible one.

Two positions are randomly chosen in the clone operator. After the two positions of the solution vector have been defined, the content of the first (relative to RB) is copied for the content of the second. Figure 5.2 shows a clone operator applied to the D2D communication RRA.

On the other hand, a position in the solution vector is randomly chosen in the random operator and the RB of that position is assigned to another randomly chosen D2D link, as shown in Figure 5.3.

Two positions are randomly chosen in the insert method. Indices 3 and 6 in Figure 5.4 were chosen as an example. The content of the smaller index is moved to the position of (larger index $+1$), i.e., the content of the smaller index is inserted in the position next to that of the larger index. Figure 5.4 displays the final solution of the aforementioned displacement and the nodes shifted (dark color) for a correct application of the rule.

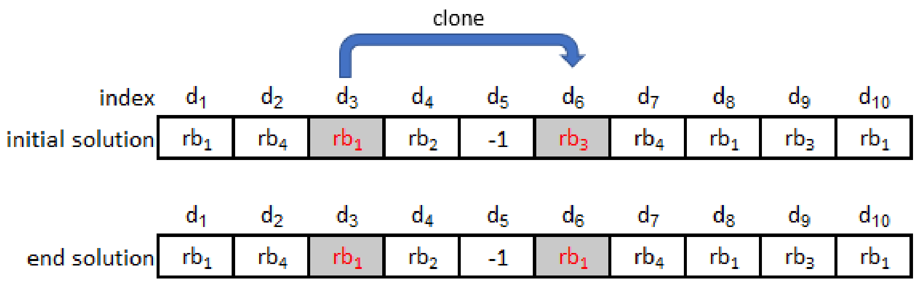


Figure 5.2: Example of a clone neighborhood structure.

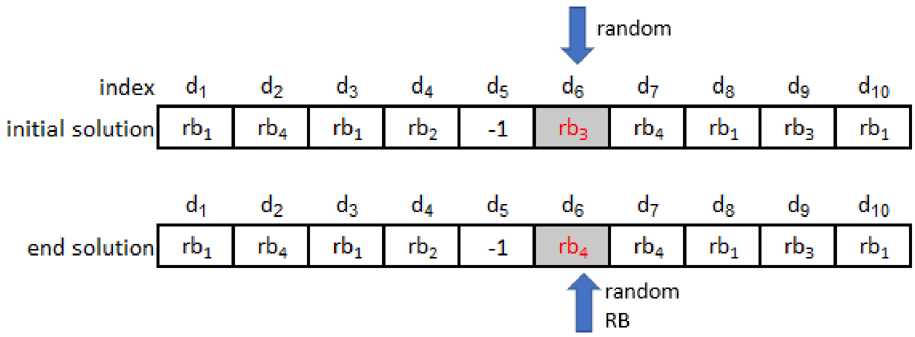


Figure 5.3: Example of a random neighborhood structure.

Similarly to the insert method, two positions are randomly chosen in the swap one. The next action is the exchange of the content of the first defined position with the content of the second one. Figure 5.5 shows an example of the swap method applied to the RRA problem.

Both employed and onlooker bees apply a self-adaptive strategy to find nearby food sources. An initial operator neighbor list (NL) of a specified length (NL_{size}) is generated by randomly populating the list with the neighboring selection approaches, as follows:

- performing a clone operator for a solution;
- performing a random operator for a solution;
- performing an insert operator for a solution;

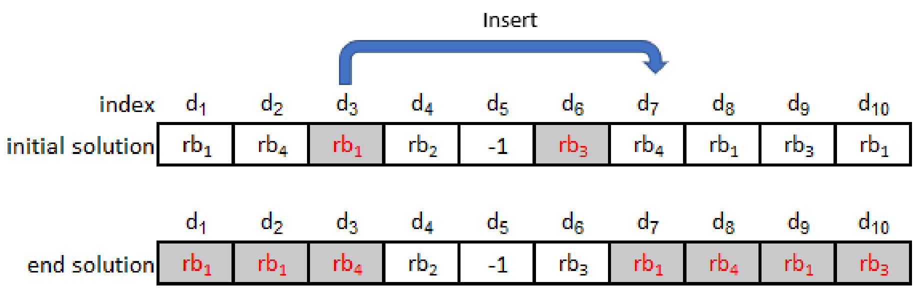


Figure 5.4: Example of an INSERT neighborhood structure.

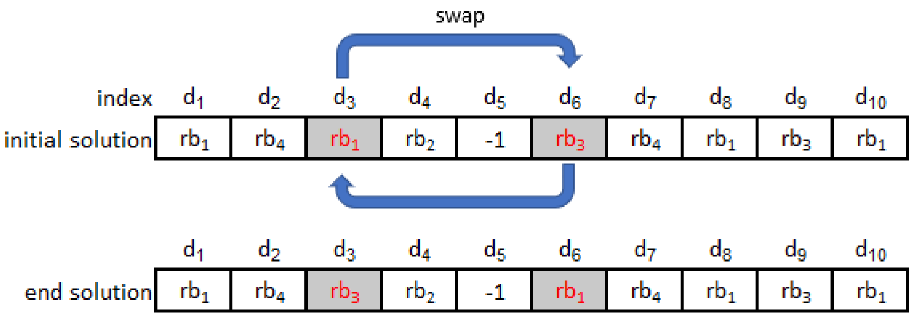


Figure 5.5: Example of a SWAP neighborhood structure.

- performing a swap operator for a solution;
- performing two clone operators for a solution;
- performing two random operators for a solution;
- performing two insert operators for a solution;
- performing two swap operators for a solution.

The computational complexity of SA-RRA-ABC for allocating D2D pairs is $\mathcal{O}(iteration\_max * SN * (du + d^2))$; since $u < d$, it is therefore $\mathcal{O}(iteration\_max * SN * d^2)$.

5.2 SECOND RRA OPTIMIZATION PROBLEM: MAXIMIZING THE ENERGY EFFICIENCY

This section is devoted to the formulation of the radio resource allocation problem for the obtaining of number of D2D communications admitted in a cellular network and maximizing the energy efficiency in the network (η_{EE}) defined as [24]

$$\eta_{EE} = \frac{R_T}{P_T} \quad [bps/Hz/J], \tag{5.5}$$

where P_T is the total power consumption in the network (Equation 3.23). A high η_{EE} is obtained with a high flow rate consuming low power. Furthermore, according to the *tradeoff* between flow rate and power consumption, the system performance metric is specified as

$$\begin{aligned} \eta_{EE}(r, P) =& \alpha_1 \sum_{i=1}^{u} r_{u_i} + \alpha_2 \sum_{j=1}^{d} r_{d_j} \\ & - \alpha_3 \sum_{i=1}^{u} P_{u_i} - \alpha_4 \sum_{j=1}^{d} P_{d_j}, \end{aligned} \tag{5.6}$$

where α_1, α_2, α_3, and α_4 are weighted parameters for the tradeoff. Two or more D2D pairs can share a same RB; an upper bound of the tolerable interference level must be considered in each RB and a minimum SINR value must be respected for cellular and D2D communications. Therefore, the optimization problem was formulated as

$$
\begin{aligned}
& \max \quad \eta_{EE}(r, P) \\
\text{s.t. } & \textit{C1: } \text{SINR}_{u_i} \geq \text{SINR}^t_{u_i}, \quad \forall u_i \in U; \\
& \textit{C2: } \text{SINR}_{d_j} \geq \text{SINR}^t_{d_j}, \quad \forall d_j \in D; \\
& \textit{C3: } x^{DM}_{d_j} + x^{RM}_{d_j} + x^{CM}_{d_j} \leq 1, \quad \forall d_j \in D; \\
& \textit{C5: } P_{u_i} \leq P^{max}_U, \quad \forall u_i \in U; \\
& \textit{C6: } (x^{DM}_{d_j} + x^{RM}_{d_j} + x^{CM}_{d_j}) P_{d_j} \leq P^{max}_D, \quad \forall d_j \in D; \\
& \textit{C7: } \sum_{j=1}^{d} x^{DM}_{d_j} + \sum_{j=1}^{d} x^{CM}_{d_j} \leq N_u; \\
& \textit{C8: } \sum_{i=1}^{u} \psi[u_i, d_j] = 1, \quad \forall d_j \in D; \\
& \textit{C9: } \psi[u_i, d_j] \in \{0, 1\}, \quad \forall d_j \in D, \quad \forall u_i \in U; \\
& \textit{C10: } x_{u_i} \in \{0, 1\}, \quad \forall u_i \in U; \\
& \textit{C11: } x_{d_j} \in \{0, 1\}, \quad \forall d_j \in D.
\end{aligned}
\tag{5.7}
$$

Constraints *C1* and *C2* guarantee the QoS requirements of cellular and D2D links, respectively, and *C3* ensures each D2D selects at most one communication mode, if admitted to the network. *C5* and *C6* guarantee the maximum transmission power requirements for cellular and D2D links, respectively. *C7* ensures the RBs used by D2D communications in dedicated and cellular modes do not exceed the number of unused RBs, *C8* guarantees each D2D pair can reuse only one RB, and *C9*, *C10*, and *C11* are numerical constraints.

The eNB is assumed to know the CSI for possible link combinations, used in the centralized resource allocation process. Three bioinspired meta-heuristics, namely (i) RRA-URLLC-ABC, (ii) RRA-URLLC-ACS-EM, and (iii) RRA-URLLC-PSO solved the RRA problem.

The solution to the RRA-URLLC problem was designed as a class consisting of 6 vectors (A1, A2, A3, A4, A5, and A6), where the index of each vector represents a cellular or D2D link. The values of A1$[1 \ldots u]$ denote the RB assigned to the cellular links. A2$[1 \ldots d]$ can take values $\{0, 1, \ldots, n, n+1, \ldots, n'\}$, where 0 represents a D2D link with no RB assigned, $\{1, \ldots, n\}$ denotes an RB assigned to the D2D link in DM or RM communication modes, and $\{n+1, \ldots, n'\}$ denotes two RBs assigned in CM. The values of A3$[1 \ldots u]$ denotes if a CUE is sharing the RB with any D2D pair or not, with a value of 1 indicating sharing. The values of A4$[1 \ldots d]$ denote the selected communication mode and those of A5$[1 \ldots u]$ and A6$[1 \ldots d]$ denote the transmission power value set for each communication. Figure 5.6 shows an example of a solution to the problem considering two CUEs, four D2D links, and five RBs.

$$
\begin{array}{ll}
A1[u] = \{1, \quad 2\} & \triangleright \text{ Cellular Allocation} \\
A2[d] = \{0, \quad 2, \quad 5, \quad <3, 4>\} & \triangleright \text{ D2D Allocation} \\
A3[u] = \{0, \quad 1\} & \triangleright \text{ Cellular Communication} \\
A4[d] = \{-1, \quad RM, \quad DM, \quad CM\} & \triangleright \text{ D2D Communication Mode} \\
A5[u] = \{P_{u_1}, \quad P_{u_2}\} & \triangleright \text{ Cellular Transmit Power} \\
A6[d] = \{0, \quad P_{d_2}, \quad P_{d_3}, \quad < P_{d_4^T}, P_{d_4^R} > \} & \triangleright \text{ D2D Transmit Power}
\end{array}
$$

Figure 5.6: Example of a solution for the RRA-URLLC problem.

$$
\begin{array}{ll}
\text{Input: } solution & \\
neighbor \leftarrow solution.clone() & \\
clink1\_id \leftarrow random(u) & \triangleright \text{ *** Cellular Link ***} \\
clink2\_id \leftarrow random(u) & \\
A5[clink1\_id] = A5[clink2\_id] & \triangleright \text{ Cellular Transmit Power} \\
 & \\
dlink1\_id \leftarrow random(d) & \triangleright \text{ *** D2D Link ***} \\
dlink2\_id \leftarrow random(d) & \\
A2[dlink1\_id] = A2[dlink2\_id] & \triangleright \text{ D2D Allocation} \\
A6[dlink1\_id] = A6[dlink12\_id] & \triangleright \text{ D2D Transmit Power}
\end{array}
$$

Figure 5.7: Example of a random neighborhood structure.

5.2.1 Artificial Bee Colony (RRA-URLLC-ABC) Algorithm

RRA-URLLC-ABC performs the sequential allocation of resources considering the RB index and, at each step, determines the CUE and the D2D pairs admitted to use the RB, according to one of three transmission power values (P_{min}, P_{mean} and P_{max}), and defines the communication mode of each D2D link. Whenever a channel has been assigned, the constraints are evaluated according to the QoS requirements of cellular communications and other admitted D2D pairs. RRA-URLLC-ABC is run on the eNB, returning the channel assigned to cellular communications, the set of admitted D2D pairs with their assigned channels, the communication mode, and the transmission power.

In the employed bees phase, each solution is updated by a neighborhood structure composed of permutation operators (random and swap). In the random operator (Figure 5.7), two positions in the solution vector are randomly chosen and the value of the first is copied into the second. In the swap operator (Figure 5.8), two positions are also randomly chosen, and the content of the first is swapped with that of the second, as in the operator used in [100]. The random operator is applied to A2, A5, and A6. It cannot be applied to A1, since two mobiles cannot use the same channel (problem constraint). The swap operator is applied to A1, A2, A5, and A6, and A3 and A4 are updated in function of the RB assignment defined in A1 and A2.

5.2.2 Ant Colony System with External Memory (RRA-URLLC-ACS-EM) Algorithm

In this study, the construction graph assigns an RB to cellular and D2D pair communications. Vertex set V_{tx} represents all possible pairs <link, RB> that exist in the scenario considered, as

```
Input: solution
neighbor ← solution.clone()
clink1_id ← random(u)                          ▷ *** Cellular Link ***
clink2_id ← random(u)

temp = A1[clink1_id]                           ▷ Cellular Allocation
A1[clink1_id] = A1[clink2_id]
A1[clink2_id] = temp

temp = A5[clink1_id]                           ▷ Cellular Transmit Power
A5[clink1_id] = A5[clink12_id]
A5[clink12_id] = temp

dlink1_id ← random(d)                          ▷ *** D2D Link ***
dlink2_id ← random(d)

temp = A2[dlink1_id]                           ▷ D2D Allocation
A2[dlink1_id] = A2[dlink2_id]
A2[dlink2_id] = temp

temp = A6[dlink1_id]                           ▷ D2D Transmit Power
A6[dlink1_id] = A6[dlink12_id]
A6[dlink12_id] = temp
```

Figure 5.8: Example of a swap neighborhood structure.

shown in Figure 5.9, whereas vertex subsets are given by all elements of V^u and V^d matrices expressed in Equations 5.8 and 5.11a, respectively.

V^u represents all possible pairs of <CUE, RB>, with the rows denoting cellular links and the columns representing RBs.

$$v^u_{i,k} = \begin{cases} 1, & \textit{if } rb_k \textit{ is allocated to } u_i, \\ 0, & otherwise. \end{cases} \tag{5.8}$$

V^d represents all possible RBs assigned (or not) to D2D links; rows correspond to D2D links and columns represent possible assignments. Note that differently to V^u, the column index starts at 0, indicating no RB was assigned to D2D communication. Therefore, $v^d_{d0} = 1$ indicates D2D pairs d_j is not admitted in the network; otherwise, $v^d_{d0} = 0$. Columns 1 to n represent the assignment of one RB in RM and DM communication modes, where

$$v^d_{j,k} = \begin{cases} 1, & \textit{if } rb_k \textit{ is allocated to } d_j, \\ 0, & otherwise. \end{cases} \tag{5.9}$$

Columns $n+1$ to n' denote two RBs allocations in CM, where

$$v^d_{j,n'} = \begin{cases} 1, & \textit{if } rb_k \textit{ and } rb_{k-1} \textit{ are allocated to } d_j, \\ 0, & otherwise, \end{cases} \tag{5.10}$$

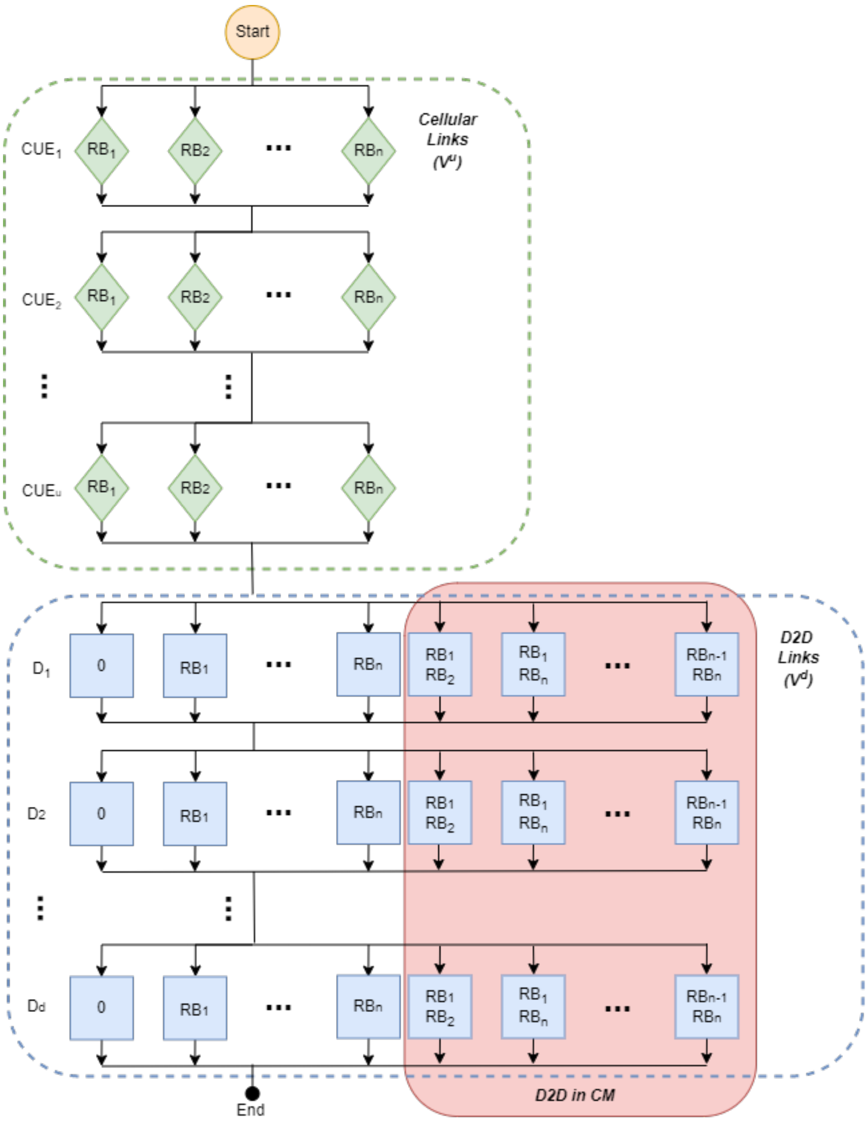


Figure 5.9: Graph for solution of RRA-URLLC problem.

$$V^d = \begin{array}{c c} & \begin{array}{cccccc} 0 & 1 & \cdots & n & \cdots & n' \end{array} \\ \begin{array}{c} 1 \\ 2 \\ \vdots \\ d \end{array} & \begin{bmatrix} v^d_{10} & v^d_{11} & \cdots & v^d_{1n} & \cdots & v^d_{1n'} \\ v^d_{20} & v^d_{21} & \cdots & v^d_{2n} & \cdots & v^d_{2n'} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ v^d_{d0} & v^d_{d1} & \cdots & v^d_{dn} & \cdots & v^d_{dn'} \end{bmatrix} \end{array}, \tag{5.11a}$$

$$n' = n + \frac{n(n-1)}{2}. \tag{5.11b}$$

5.2.3 RRA-URLLC-PSO Algorithm

Crisp set E is formed by elements of V^u and V^d matrices. The set with possibilities A_p represents a row in V^u or V^d, depending on the link being evaluated at each step of the solution construction.

Similarly to ACS-EM algorithm, when the particle position is updated, a transmission power value is randomly selected between P_{min} and P_{max}. At each new position, the communication mode is updated. A solution such as the one shown in Figure 5.6 is associated with each particle; velocities and positions selected by the particles are updated according to the pseudocode presented in Algorithm 3.

5.3 FINAL CONSIDERATIONS

This chapter discussed the two proposed approaches to the RRA problem. The first formulates the RRA problem as an optimization problem that maximizes the weighted system throughput. The aim is to obtain the number of D2D communications admitted into an network. The objective function considers both the social and physical layers, where D2D communications with higher intensity of social relationship between transmitter and receiver are more probable to be accepted in the network. D2D and cellular communications share resources using reuse mode and interference is managed. The problem is solved by the SA-RRA-ABC meta-heuristic and its modeling considered a partial CSI model with limited feedback.

The second approach formulates the RRA problem as an optimization problem that maximizes the energy efficiency of the network. The chapter presents the formulation of the problem to obtain the RBs assignment to cellular and D2D communications, the transmission power of the cellular and D2D links, and the communication mode of each D2D link (three communication modes DM, RM and CM are considered). A description of the three meta-heuristics, used to solve the problem, are presented at the end of the chapter. They were RRA-URLLC-ABC, RRA-URLLC-ACS-EM, and RRA-URLLC-PSO.

6 EVALUATION AND DISCUSSION

This chapter provides the results of the approaches proposed in this research to CPP and RRA. The meta-heuristics used to solve the problems are compared and the stopping criterion used in all experiments is the number of iterations. The chapter is divided into three sections.

In the first section, we present the evaluation of the proposed solution to the CPP and the two meta-heuristics proposed for the solution of the CPP problem are compared.

In the second section, we present the evaluation of the first proposed RRA optimization problem by comparing the proposed SA-RRA-ABC solution with the greedy solutions used in the first step of the proposed CPP solution. In this session the impact of the feedback model parameters on the weighted system throughput and access rate is evaluated.

In the third section, we present the evaluation of the second proposed RRA optimization problem and study the impact of the parameters associated with the services URLLC (blocklength and decoding error probability) on the system throughput. Moreover, the impact of the objective function weights on the various metrics (number of D2D admitted to the network, system throughput, power consumption, and network energy efficiency) is evaluated and the three meta-heuristics proposed for the solution to the RRA-URLLC problem are compared.

6.1 CPP EVALUATION

This section provides the results of the approach proposed in this research to CPP, considering D2D communications, that take place in an LTE-A network. In the first step, two greedy heuristics are used to obtain the number of D2D communications allowed in the network (C_{D2D}). One algorithm is social-unaware, considering the communications that generate less interference, and the other is social-aware, considering the intensity of the social relationship between users.

In the second step, the ABC, ACS-EM, ACS [101] and Particle Swarm Optimization (PSO) [112] meta-heuristics are applied to solve CPP. The goal to find the number and respective location of SDN controllers, necessary for the management of D2D communications occurring in an LTE-A cellular network, as well as the eNB-controller assignment.

An LTE-A network with 1 PGW, 2 SGW and 8 eNBs was considered for the evaluation of the influence of D2D communications on the location and number of controllers. Table 6.1 shows the parameters of the System Model.

Some adjustments were also made in the simulation environment for the definition of the values of the ABC and the ACS-EM meta-heuristics parameters. The best values were:

(i) for ABC: $SN = 100$; $n_0 = 0.5$; $limit = 5$ and $NL_{size} = 5$; and

(ii) for ACS-EM: $ant\_quantity = 100$; $q_0 = 0.2$; $r_0 = 0.2$; $z_0 = 0.2$; $\alpha = 0.1$; $\beta = 0.9$; $\tau_0 = 0.05$ and $\delta = 0.1$.

The meta-heuristics were applied 32 times with the same parameters and 450 iterations were considered.

Table 6.1: Simulation parameters

Parameter	Value
Carrier frequency	2 GHz [28, 75]
system bandwidth - W	1 MHz
Number of CUEs	10 user per eNB
DUE density	$10^{-3} user/Km^2$
CUE transmitted power	23 dBm [75]
DUE transmitted power	20 dBm [75]
eNB coverage radius	500 m [75]
Distance between D2D pair	Uniformly distributed in [0,20]m
t_{QoS}	4 ms
D2D request rate - λ	0.1 req/ms
core processing rate - μ	50 req/ms
number of controller cores - m	4
SDN controller capacity - K	4μ
$\lambda_{lost}^{threshold}$	5% of $c \cdot \lambda_c$
Path loss model for cellular link	$128.1 + 37.6 log_{10}(d[Km])$ [75]
Path loss model for D2D pair	$148 + 40 log_{10}(d[Km])$ [75]
Shadow fading standard deviation	10 dB for cellular link, 12 dB for D2D pair
Small-scale fading	i.i.d complex Gaussian distributed with zero mean and unit variance
Antenna gain	14 dBi for eNB [75] 0 dBi for UE [75]
single-sided noise power spectral density - N_0	-174 dBm/Hz [28, 75]
SINR$_u^t$	10 dB
SINR$_d^t$	10 dB [75]
number of iterations	450
γ_c	0.4
γ_π	0.1
γ_T	0.3
γ_F	0.2

Different values of λ were used for the evaluation of the effect of an increase/decrease of control traffic related to D2D communications. In Table 6.2 we present the results obtained in the social-unaware scenario when the ACS-EM is used. In Table 6.3 we present the results obtained in the social-aware scenario when the ABC is used. The following metrics were obtained: number of controllers, response time, use of average controllers, assignment of eNB-controllers, location of controllers in the cellular network, and lost request rate in the SDN controller.

According to Tables 6.2 and 6.3, the number of controllers required increases only when the constraints are not satisfied. As an illustrative example, in Table 6.2 two controllers are used for

$\lambda = 0.4\ req/ms$; however, the controller load is $\simeq 100\%$, and the loss rate begins to increase. When λ increases to $0.5\ req/ms$, more controllers are used not to exceed the threshold set for lost requests rate and satisfy all constraints of the problem.

Table 6.2: Simulation Results of social-unaware scenario using ACS-EM algorithm

$\lambda\ (req/ms)$	Nro Controller	Tr	Controller Utilization	eNB-Controller	Controller Location	$\lambda_{lost}(req/ms)$
0.1	1	1.525	54.78 %	$b_0c_0, b_1c_0, b_2c_0, b_3c_0,$ $b_4c_0, b_5c_0, b_6c_0, b_7c_0$	$c_0 - SGW_0$	0
0.2	2	0.9594	54.78 %	b_0c_0, b_4c_0, b_6c_0 $b_1c_1, b_2c_1, b_3c_1, b_5c_1, b_7c_1$	$c_0 - eNB_0$ $c_1 - SGW_1$	0
0.3	2	1.2132	81.84 %	b_3c_0, b_5c_0, b_7c_0 $b_0c_1, b_1c_1, b_2c_1, b_4c_1, b_6c_1$	$c_0 - eNB_5$ $c_1 - eNB_4$	0.5813
0.4	2	2.1907	100 %	$b_0c_0, b_2c_0, b_6c_0, b_7c_0$ $b_1c_1, b_3c_1, b_4c_1, b_5c_1$	$c_0 - eNB_0$ $c_1 - eNB_5$	17.2
0.5	3	1.1061	90.79 %	b_0c_0, b_2c_0, b_7c_0 b_1c_1, b_3c_1, b_5c_1 b_4c_2, b_6c_2	$c_0 - eNB_0$ $c_1 - eNB_5$ $c_2 - eNB_6$	0.9102
0.6	4	0.5432	82.17 %	b_3c_0, b_5c_0 b_2c_1, b_4c_1 b_0c_2, b_6c_2 b_1c_3, b_7c_3	$c_0 - eNB_3$ $c_1 - eNB_2$ $c_2 - eNB_0$ $c_3 - eNB_1$	0
0.7	4	0.7518	95.73 %	b_3c_0, b_5c_0 b_2c_1, b_4c_1 b_1c_2, b_7c_2 b_0c_3, b_6c_3	$c_0 - eNB_3$ $c_1 - eNB_4$ $c_2 - eNB_1$ $c_3 - eNB_6$	0.2344
0.8	4	1.2379	98.67 %	b_1c_0, b_3c_0 b_2c_1, b_6c_1 b_5c_2, b_7c_2 b_0c_3, b_4c_3	$c_0 - eNB_1$ $c_1 - eNB_2$ $c_2 - eNB_5$ $c_3 - eNB_0$	19.6003
0.9	6	0.4007	79.07 %	b_7c_0 b_6c_1 b_4c_2 b_5c_3 b_1c_4, b_3c_4 b_0c_5, b_2c_5	$c_0 - eNB_7$ $c_1 - eNB_6$ $c_2 - eNB_4$ $c_3 - eNB_5$ $c_4 - eNB_3$ $c_5 - eNB_0$	5.5821
1	7	0.2384	76.9 %	b_7c_0 b_3c_1 b_0c_2, b_2c_2 b_1c_3 b_4c_4 b_6c_5 b_5c_6	$c_0 - eNB_7$ $c_1 - eNB_3$ $c_2 - eNB_0$ $c_3 - eNB_1$ $c_4 - eNB_4$ $c_5 - eNB_6$ $c_6 - eNB_5$	2.4286

As shown in Figure 6.1, the increase in the rate of requests generated by D2D communications increases the number of required SDN controllers. It is also shown that in the social-aware scenario, a smaller number of controllers is needed. This result is obtained since the social-aware heuristic makes the greedy choice based on the intensity of the social relationship between D2D pairs, resulting in a smaller number of D2D communications admitted in eNBs. In the heuristic (social-unaware), the greedy choice is considering the D2D pairs with the least interference factor

Table 6.3: Simulation Results of social-aware scenario using ABC algorithm

$\lambda\ (req/ms)$	Nro Controller	Tr	Controller Utilization	eNB-Controller	Controller Location	$\lambda_{lost}(req/ms)$
0.1	1	1.5223	21.89 %	$b_0c_0, b_1c_0, b_2c_0, b_3c_0,$ $b_4c_0, b_5c_0, b_6c_0, b_7c_0$	$c_0 - SGW_0$	0
0.2	1	1.5234	43.78 %	$b_0c_0, b_1c_0, b_2c_0, b_3c_0,$ $b_4c_0, b_5c_0, b_6c_0, b_7c_0$	$c_0 - SGW_0$	0
0.3	1	1.5281	65.67 %	$b_0c_0, b_1c_0, b_2c_0, b_3c_0,$ $b_4c_0, b_5c_0, b_6c_0, b_7c_0$	$c_0 - SGW_0$	0
0.4	1	1.5552	87.56 %	$b_0c_0, b_1c_0, b_2c_0, b_3c_0,$ $b_4c_0, b_5c_0, b_6c_0, b_7c_0$	$c_0 - SGW_1$	0
0.5	2	0.9583	54.72 %	b_3c_0, b_5c_0, b_7c_0 $b_0c_1, b_1c_1, b_2c_1, b_4c_1, b_6c_1$	$c_0 - eNB_7$ $c_1 - eNB_6$	0
0.6	2	0.9614	65.67 %	b_3c_0, b_5c_0, b_7c_0 $b_0c_1, b_1c_1, b_2c_1, b_4c_1, b_6c_1$	$c_0 - eNB_3$ $c_1 - eNB_2$	0
0.7	2	0.9683	76.61 %	$b_0c_0, b_1c_0, b_2c_0, b_4c_0, b_6c_0$ b_3c_1, b_5c_1, b_7c_1	$c_0 - eNB_6$ $c_1 - eNB_3$	0
0.8	2	0.9886	87.56 %	$b_0c_0, b_1c_0, b_2c_0, b_4c_0, b_6c_0$ b_3c_1, b_5c_1, b_7c_1	$c_0 - eNB_6$ $c_1 - eNB_5$	0
0.9	2	1.247	98.37 %	$b_0c_0, b_1c_0, b_2c_0, b_4c_0, b_6c_0$ b_3c_1, b_5c_1, b_7c_1	$c_0 - eNB_6$ $c_1 - eNB_3$	0.2275
1	2	1.8743	100 %	b_3c_0, b_5c_0, b_7c_0 $b_0c_1, b_1c_1, b_2c_1, b_4c_1, b_6c_1$	$c_0 - eNB_5$ $c_1 - SGW_0$	17

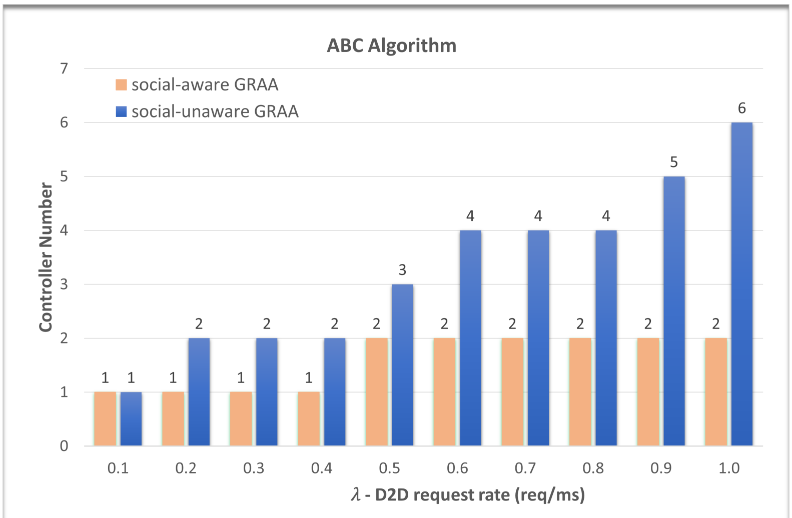
resulting in a higher number of D2D communications admitted in the eNBs.

The results also show when more than one controller is used, their best location is in the access network, near D2D communications, since eNBs represent the first hop, which reduces both propagation latency and total response time. However, when only one controller is used, it must be placed in the core of the network, in the SGW entity, which increases the response time of each eNB for the controller. Regarding the entire network, the average value of the response time provides a better result without exceeding the QoS value considered.

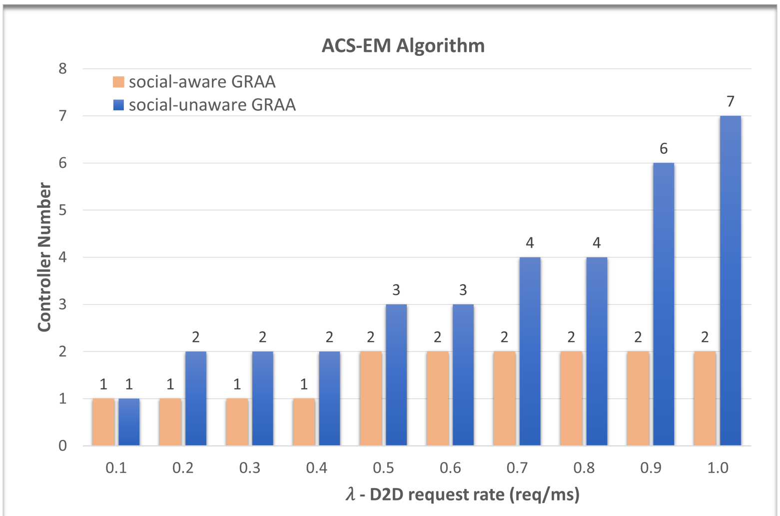
The fairness in the allocation of the control for the eNB is also among the objectives. As shown in Tables 6.2 and 6.3, the assignment was balanced between the controllers; therefore, the results from ABC and ACS-EM are consistent with the expected solutions.

ABC and ACS-EM were compared with the ACS [101] and the PSO [112], for the evaluation of its effectiveness. ACS was applied with the following parameters: $ant\_quantity = 100$; $q_0 = 0.2$; $\beta = 2$; $\tau_0 = 0.05$ and $\delta = 0.1$.

In PSO [112], the particles cooperate to search for the global optimum in the n-dimensional search space. The *ith* particle maintains both position and velocity. In each iteration, each particle uses its own search experience (self-cognitive) and the whole swarm's search experience (social-influence) to update velocity, and flies to a new position. Therefore, $c1$ and $c2$ are two parameters that weight the importance of self-cognitive and social-influence, respectively. A scheme that



(a) ABC Algorithm



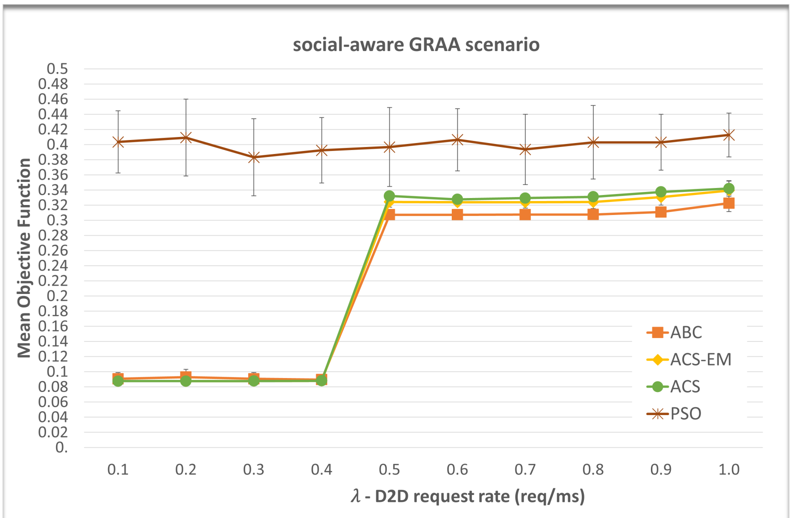
(b) ACS-EM Algorithm

Figure 6.1: Effect of λ on the number of required SDN controllers.

linearly decrements inertia weight w between w_{max} and w_{min} is used in [112] to update the velocity. The discrete PSO modeling presented in [112] was applied with the following parameters: $particle\_quantity = 100$; $w_{max} = 0.9$; $w_{min} = 0.4$; $c1 = 2$; $c2 = 2$.

Figure 6.2 shows the average of the objective function value obtained in the 32 observations. The standard deviation of the results for ABC, ACS-EM and ACS is small, whereas the results for PSO show a higher dispersion. Since ABC achieved a lower average in most instances, we inferred its performance was probably better than that of ACS-EM, ACS and PSO meta-heuristics. However, statistical tests were applied for theoretically supporting the previous statement.

(a) social-unaware scenario



(b) social-aware scenario

Figure 6.2: Algorithms Comparison - Average Objective Function.

First, the nature of the samples was tested for checking their distribution (normal or not). If the results followed a normal distribution, parametric hypothesis tests were applied for finding the one of better performance. Otherwise, non-parametric hypothesis tests were applied. A 95%

confidence level was used for all tests.

The experiments were conducted according to the following methodology:

1. search the results for ABC, ACS-EM, ACS and PSO using the system model defined at the beginning of this section; 32 simulations (observations) were performed for each case;

2. generate the three samples with the best results for each case (32 observations);

3. verify if the results of each meta-heuristic follow a normal distribution using the Kolmogorov-Smirnov test [113];

4. apply the Kruskal-Wallis test [114] among the meta-heuristics samples to assess whether or not their population medians differ (whether three independent samples had been selected from populations with an identical distribution).

5. apply Wilconxon test [113, 114] among the meta-heuristic of best median and all others to find the best.

As shown in Tables 6.4 and 6.5, the results of the Kolmogorov-Smirnov test for the three meta-heuristics were $h = 1$, which rejects the null hypotheses. Therefore, the samples obtained do not follow a normal distribution, and non-parametric tests must be used.

Kruskal-Wallis test was applied to determine whether at least one meta-heuristic showed a different median and also whether a performance comparison could be made between them. The null hypothesis (Ho) assumes independent samples from two or more groups provide distributions with equal medians. If $h = 1$, the null hypothesis is rejected. Therefore, the meta-heuristic of lowest median should be compared with the others by Wilcoxon test for proving its best performance in solving the CPP problem.

Wilcoxon Rank Sum was applied to find the meta-heuristic of best performance. The null hypothesis (Ho) assumes the two independent samples derive from distributions with equal medians. If $h = 1$, the null hypothesis is rejected. Therefore, the meta-heuristic of lower median shows better performance. Otherwise, no conclusion can be drawn.

The Wilcoxon test result was $h = 1$ for eight instances of the problem, and ABC showed a lower median, which indicates better performance (see Tables 6.4 and 6.5). No conclusions could be drawn between the ABC and ACS-EM meta-heuristics for some values of λ, as for example in the social-unaware scenario for the values $\lambda = 0.7$ and $\lambda = 0.9$ (see Table 6.5), which means that they perform similarly in these cases. However, since ABC performed better than PSO for all instances and better than the ACS-EM and ACS meta-heuristics in most instances, we can conclude its performance is higher in solving the SDN controller placement problem in D2D communications management for the investigated scenario.

The computational complexities of the four meta-heuristics were analyzed (more details are shown in Appendix A), and are expressed as

Table 6.4: Hypothesis Test Results social-aware scenario

$\lambda\,(req/ms)$	Algorithm	Kolmogorov-Smirnov Test (h)	Wilcoxon Rank Sum Test (h)	Median
0.1	ABC	1.0	ABC - ACS-EM => 1.0 ABC - ACS => 1.0 ABC - PSO => 1.0	0.0876
	ACS-EM	1.0		0.0876
	ACS	1.0		0.0876
	PSO	1.0		0.3995
0.2	ABC	1.0	ABC - ACS-EM => 1.0 ABC - ACS => 1.0 ABC - PSO => 1.0	0.0876
	ACS-EM	1.0		0.0876
	ACS	1.0		0.0876
	PSO	1.0		0.4182
0.3	ABC	1.0	ABC - ACS-EM => 1.0 ABC - ACS => 1.0 ABC - PSO => 1.0	0.0877
	ACS-EM	1.0		0.0877
	ACS	1.0		0.0877
	PSO	1.0		0.3958
0.4	ABC	1.0	ABC - ACS-EM => 0.0 ABC - ACS => 0.0 ABC - PSO => 1.0	0.0880
	ACS-EM	1.0		0.0880
	ACS	1.0		0.0880
	PSO	1.0		0.3981
0.5	ABC	1.0	ABC - ACS-EM => 1.0 ABC - ACS => 1.0 ABC - PSO => 1.0	**0.3074**
	ACS-EM	1.0		0.3260
	ACS	1.0		0.3311
	PSO	1.0		0.4075
0.6	ABC	1.0	ABC - ACS-EM => 1.0 ABC - ACS => 1.0 ABC - PSO => 1.0	**0.3074**
	ACS-EM	1.0		0.3256
	ACS	1.0		0.3299
	PSO	1.0		0.4173
0.7	ABC	1.0	ABC - ACS-EM => 1.0 ABC - ACS => 1.0 ABC - PSO => 1.0	**0.3075**
	ACS-EM	1.0		0.3279
	ACS	1.0		0.3311
	PSO	1.0		0.4070
0.8	ABC	1.0	ABC - ACS-EM => 1.0 ABC - ACS => 1.0 ABC - PSO => 1.0	**0.3076**
	ACS-EM	1.0		0.3255
	ACS	1.0		0.3315
	PSO	1.0		0.4112
0.9	ABC	1.0	ABC - ACS-EM => 1.0 ABC - ACS => 1.0 ABC - PSO => 1.0	**0.3100**
	ACS-EM	1.0		0.3377
	ACS	1.0		0.3383
	PSO	1.0		0.4030
1.0	ABC	1.0	ABC - ACS-EM => 1.0 ABC - ACS => 1.0 ABC - PSO => 1.0	**0.3186**
	ACS-EM	1.0		0.3464
	ACS	1.0		0.3464
	PSO	1.0		0.4078

Table 6.5: Hypothesis Test Results social-unaware scenario

$\lambda\ (req/ms)$	Algorithm	Kolmogorov-Smirnov Test (h)	Wilcoxon Rank Sum Test (h)	Median
0.1	ABC	1.0	ABC - ACS-EM => 1.0 ABC - ACS => 1.0 ABC - PSO => 1.0	0.0876
	ACS-EM	1.0		0.0876
	ACS	1.0		0.0876
	PSO	1.0		0.3924
0.2	ABC	1.0	ABC - ACS-EM => 1.0 ABC - ACS => 1.0 ABC - PSO => 1.0	**0.2935**
	ACS-EM	1.0		0.3226
	ACS	1.0		0.3338
	PSO	1.0		0.4122
0.3	ABC	1.0	ABC - ACS-EM => 1.0 ABC - ACS => 1.0 ABC - PSO => 1.0	**0.2936**
	ACS-EM	1.0		0.3226
	ACS	1.0		0.3334
	PSO	1.0		0.4103
0.4	ABC	1.0	ABC - ACS-EM => 1.0 ABC - ACS => 1.0 ABC - PSO => 1.0	**0.3371**
	ACS-EM	1.0		0.3876
	ACS	1.0		0.4034
	PSO	1.0		0.4083
0.5	ABC	1.0	ABC - ACS-EM => 1.0 ABC - ACS => 1.0 ABC - PSO => 1.0	**0.3879**
	ACS-EM	1.0		0.3902
	ACS	1.0		0.3942
	PSO	1.0		0.4065
0.6	ABC	1.0	ABC - ACS-EM => 1.0 ABC - ACS => 1.0 ABC - PSO => 1.0	**0.4062**
	ACS-EM	1.0		0.4127
	ACS	1.0		0.4277
	PSO	1.0		0.4343
0.7	ABC	1.0	ABC - ACS-EM => 0.0 ABC - ACS => 1.0 ABC - PSO => 1.0	**0.4079**
	ACS-EM	1.0		0.4085
	ACS	1.0		0.4100
	PSO	1.0		0.4372
0.8	ABC	1.0	ABC - ACS-EM => 1.0 ABC - ACS => 1.0 ABC - PSO => 1.0	**0.4102**
	ACS-EM	1.0		0.4110
	ACS	1.0		0.4229
	PSO	1.0		0.4304
0.9	ABC	1.0	ABC - ACS-EM => 0.0 ABC - ACS => 0.0 ABC - PSO => 1.0	0.4417
	ACS-EM	1.0		0.4417
	ACS	1.0		0.4417
	PSO	1.0		0.4499
1.0	ABC	1.0	ABC - ACS-EM => 1.0 ABC - ACS => 1.0 ABC - PSO => 1.0	**0.4219**
	ACS-EM	1.0		0.4442
	ACS	1.0		0.4442
	PSO	1.0		0.4578

$$
\begin{aligned}
&ABC => O(iteration\_max * [SN * (SNlogSN + \theta)]), && (6.1a)\\
&ACS\text{-}EM => O(iteration\_max * [ant\_quantity * b\theta + c\theta^2]), && (6.1b)\\
&ACS => O(iteration\_max * [ant\_quantity * b\theta + c\theta^2]), && (6.1c)\\
&PSO => O(iteration\_max * particle\_quantity * c\theta). && (6.1d)
\end{aligned}
$$

Figure 6.3 displays the number of computations in respect to the agents quantity. In ACS-EM and ACS, the agents (ants) need to go through each node of the graph in each iteration to build a new solution. Differently in ABC, the agents (bees) use a neighborhood structure where some nodes are selected to explore neighbor solutions, with a smaller number of operations for the exploring of a new solution. Similar to ABC, the agents (particles) in PSO explore new solutions by changing the position and speed of the particles without having to go through all the nodes considered in a feasible solution. So, the number of operations performed by the agents in ABC and PSO is less than the operations performed by the agents of the ACS-EM and ACS that need to go through all nodes. As a consequence, the increase in the number of agents in ACS-EM and ACS causes a rapid growth in the number of computations while this increase is smaller in ABC and PSO. Therefore, we can conclude that the ABC and PSO meta-heuristics deliver solutions in less time and with lower computational cost.

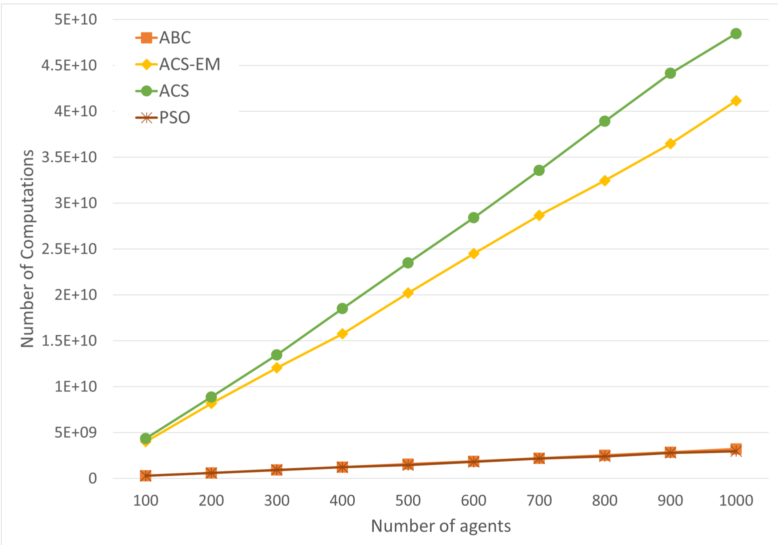


Figure 6.3: Computational complexity versus numbers of agents.

Figure 6.4 displays the computational complexity (number of operations performed in the execution of the meta-heuristics) in respect to the eNBs quantity. We can observe that as the size of the network increases, the computational complexity of the meta-heuristics increases. This increase is polynomial in ACS-EM and ACS, a result consistent with the computational complexity calculated in the equations 6.1b and 6.1c where the number of operations is affected by the square of the set Θ (CPP - set of possible locations of the controllers). The results show that PSO has the lowest computational complexity and its increase is less significant when the number of eNBs increases. In the sequence, the ABC showed less computational complexity compared to ACS-EM and ACS, presenting a logarithmic growth with an increase in the number of eNBs, a result consistent with the formulation presented in the equation 6.1a.

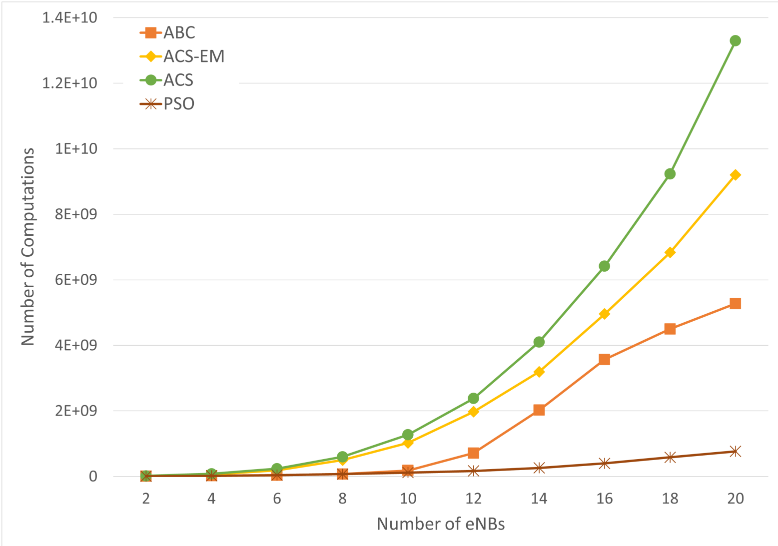


Figure 6.4: Computational complexity versus number of eNBs.

The analysis revealed the computational complexity of ABC and PSO are lower than that of ACS-EM and ACS. However, ABC and ACS-EM showed better performance in solving the problem, according to statistical tests, with characteristics that enable a more efficient exploration of the search space, thus, avoiding sub-optimal solutions.

Due to the focus of external memory addressed in ACS-EM, in certain cases, ants do not make random decisions, and choose solution components influenced by the values recorded in that memory. Since it stores information specific to the search history from the beginning of the algorithm, it effectively focuses on regions of the search space not visited (not recorded in memory and with values reflecting this), or, instead, on those already visited and that are promising (also recorded in memory and with values that reflect it). Such uses of memory reflect the intensification and diversification mechanisms that avoid premature convergence and achieve higher performance [14] against optimization problems of particular characteristics, such as the one discussed in this article.

Similarly happens with ABC that favors searches in neighborhoods of known solutions using self-adaptive strategy. Similarly with the memory-enabled ants, the worker bees memorize a food source and explore its neighborhood, maintaining a list of winning approaches. Using the winning operators it is possible to find neighbors who improve the food source more often, reducing exploration operations. In this sense, ABC intensification is more favored than would be desired, having a greater influence when the cardinality of the set of feasible solutions increases, since the degree of exploration of the ABC decrease. This effect can be seen in Figure 6.2(b) where for values of $\lambda \leq 0.4\, req/ms$ the meta-heuristics ACS and ACS-EM manage to deliver better results of the objective function.

According to the results, ABC and ACS-EM perform better in the CPP solution, and in most scenarios considered, the ABC meta-heuristic performed better than the ACS-EM. Also, the computational complexity of ABC is lower, which makes the meta-heuristic more attractive in time sensitive scenarios, such as dynamic scenarios. Therefore, the ABC is better than the other meta-heuristics evaluated in the CPP solution, for the investigated scenarios.

6.2 FIRST RRA OPTIMIZATION PROBLEM

This section provides the results of the approach for the social-aware RRA problem, which considers D2D communications established in an LTE-A network. A cellular LTE-A network is simulated with 1 PGW, 1 SGW, and 1 eNB, and a Twitter dataset [115] is used for the determination of the social relationships between DUEs devices. Simulation results characterize the cell throughput with partial CSI and gain of quantitative insights into its behavior. The numerical results are averaged over 100 user drops. In each drop, U CUEs and the d_j^R of D D2D pairs are randomly located with uniform probability in a cell of $500m$ radius. The location of d_j^T is chosen for the modeling of different d_j^T - d_j^R distances, so that it lies randomly with uniform probability within a circle of $50m$ radius around d_j^R. Table 6.6 shows the parameters of the System Model.

Table 6.6: Simulation parameters

Parameter	Value
Carrier frequency	2 GHz [75]
Bandwidth of component carrier	1.4 MHz per subchannel @ 6 RBs (72 subcarriers)
CUE transmitted power	10 dBm
DUE transmitted power	-10 dBm
eNB coverage radius	500 m
Distance between D2D pairs	Uniformly distributed in [0,50]m
Path loss model for cellular link	$128.1 + 37.6 log_{10}(l[Km])$
Path loss model for D2D pairs	$148 + 40 log_{10}(l[Km])$
lognormal shadowing fading standard deviation	10 dB for cellular link; 12 dB for D2D pair link
Small-scale fading	Rayleigh model
Antenna gain	14 dBi for eNB; 0 dBi for UE
single-sided noise power spectral density - N_0	-174 dBm/Hz
SINR$_u^t$	10 dB
SINR$_d^t$	10 dB
number of iterations	250

The simulation environment involves programs written in C# for the definition of the values of the Social-aware RRA Artificial Bee Colony (SA-RRA-ABC) parameters. The values considered were $SN = 25$, $limit = 5$, and $NL_{size} = 10$.

The performance of SA-RRA-ABC was benchmarked with the following schemes:

- Exhaustive Search (ES): all possible assignments of D2D pairs to subchannels were searched towards finding the optimal solution, which is computationally infeasible, except for small values of u and d.
- GRAA, proposed in [75]: greedy heuristics consider a sequential resource allocation mode, where eNB determines the admitted D2D pairs and allocates resources based on the sequential indexing of RB. Conflict graphs are updated whenever a D2D pair has been selected to use a resource block, and tolerable interference values are evaluated by both eNB and other DUEs admitted. In GRAA algorithm, D2D pairs are selected according to the degree of interference. Therefore, the D2D pairs of lowest degree are chosen first. Such a greedy solution aims to maximize the total admitted D2D pairs; however, it does not consider the system throughput.
- SA-GRAA: an update of GRAA, it considers the social relationship between users in the greedy D2D link selection.

In SA-GRAA, the selection of D2D pairs is based on the intensity of the social relationships (ω_{d_j}) between DUEs. Therefore, D2D pairs of higher intensity are selected first. SA-GRAA, SA-GRAA and GRAA were benchmarked with ES for different n, M, and q feedback parameters. For $q \geq 4$, it becomes computationally cumbersome to numerically optimize $L = 2^q$ threshold levels; therefore, the threshold values shown in Table 6.7 were used as system parameters. They start at 10 dB, which is the SINR threshold for UEs and has 5 dB increments in the quantization scheme.

Table 6.7: D2D SINR Quantization for Different q

q	D2D SINR in dB (SINR_i)
0	$\text{SINR}_0 = 10$ dB
1	$\text{SINR}_0 = 10$ dB, $\text{SINR}_1 = 15$ dB
2	$\text{SINR}_0 = 10$ dB, $\text{SINR}_1 = 15$ dB, $\text{SINR}_2 = 20$ dB, $\text{SINR}_3 = 25$ dB
3	$\text{SINR}_0 = 10$ dB, $\text{SINR}_1 = 15$ dB, $\text{SINR}_2 = 20$ dB, $\text{SINR}_3 = 25$ dB, $\text{SINR}_4 = 30$ dB, $\text{SINR}_5 = 35$ dB, $\text{SINR}_6 = 40$ dB, $\text{SINR}_7 = 45$ dB
4	$\text{SINR}_0 = 10$ dB, $\text{SINR}_1 = 15$ dB, $\text{SINR}_2 = 20$ dB, $\text{SINR}_3 = 25$ dB, $\text{SINR}_4 = 30$ dB, $\text{SINR}_5 = 35$ dB, $\text{SINR}_6 = 40$ dB, $\text{SINR}_7 = 45$ dB $\text{SINR}_8 = 50$ dB, $\text{SINR}_9 = 55$ dB, $\text{SINR}_{10} = 60$ dB, $\text{SINR}_{11} = 65$ dB, $\text{SINR}_{12} = 70$ dB, $\text{SINR}_{13} = 75$ dB, $\text{SINR}_{14} = 80$ dB, $\text{SINR}_{15} = 85$ dB

Since ES has $\mathcal{O}(d^u)$ computational complexity when full CSI and $\mathcal{O}(M^N)$) are considered for the feedback model, it requires considerable computational effort for high values of $N \leq u$ and $M \leq d$. Therefore, the evaluation involved a small scenario with $u = 5$ and $d = 10$ pairs towards results with reasonable effort.

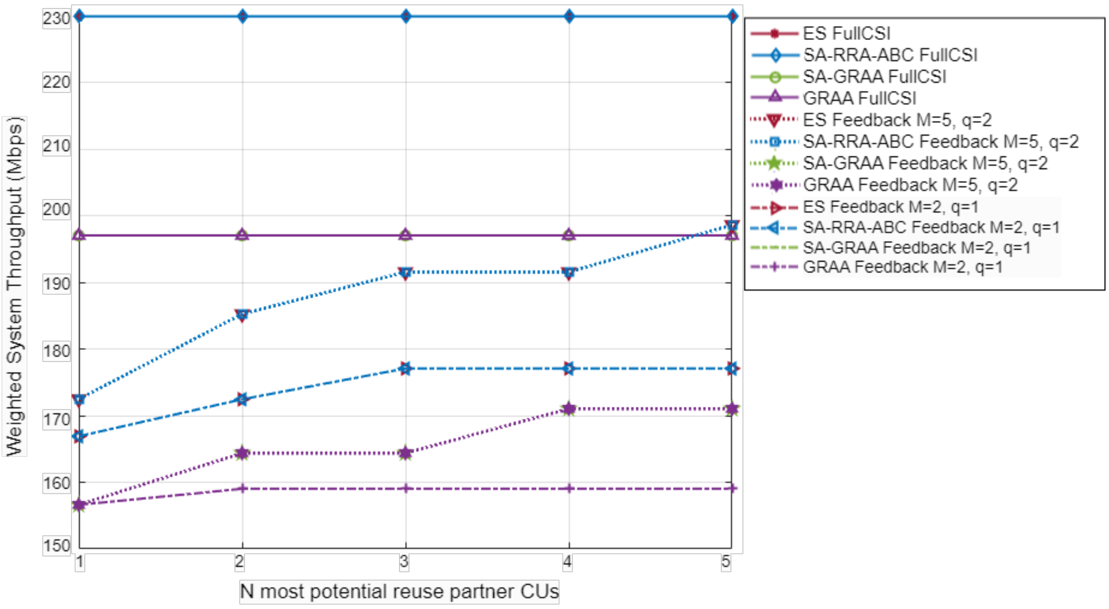
The following two metrics were used for the performance evaluation: D2D access rate, defined as the ratio of number of accessed D2D pairs and total D2D pairs, and weighted system throughput, which is the sum of the throughput of each CUE and the D2D pairs weighted by the intensity of the social relationship between them. In weighted system throughput, D2D pairs with stronger social relationship have higher throughput. Both weighted system throughput and access rate are constant when full CSI is considered and represent the upper limit of comparison. In this scenario, eNB always knows the instantaneous CSI for all links, and, consequently, a change in the feedback parameters does not affect the result.

Figure 6.5 shows a comparison of the weighted system throughput and access rate of SA-RRA-ABC, SA-GRAA, GRAA and ES when the number of potential sub-channels, N, varies in the feedback scheme. SA-RRA-ABC can find the optimal solution achieving the same results of ES, while greedy solutions show lower performance. Greedy solutions provided similar results, therefore, no considerable gain was obtained in SA-GRAA – the greedy choice was changed towards selecting D2D pairs with greater strength of social relationships.

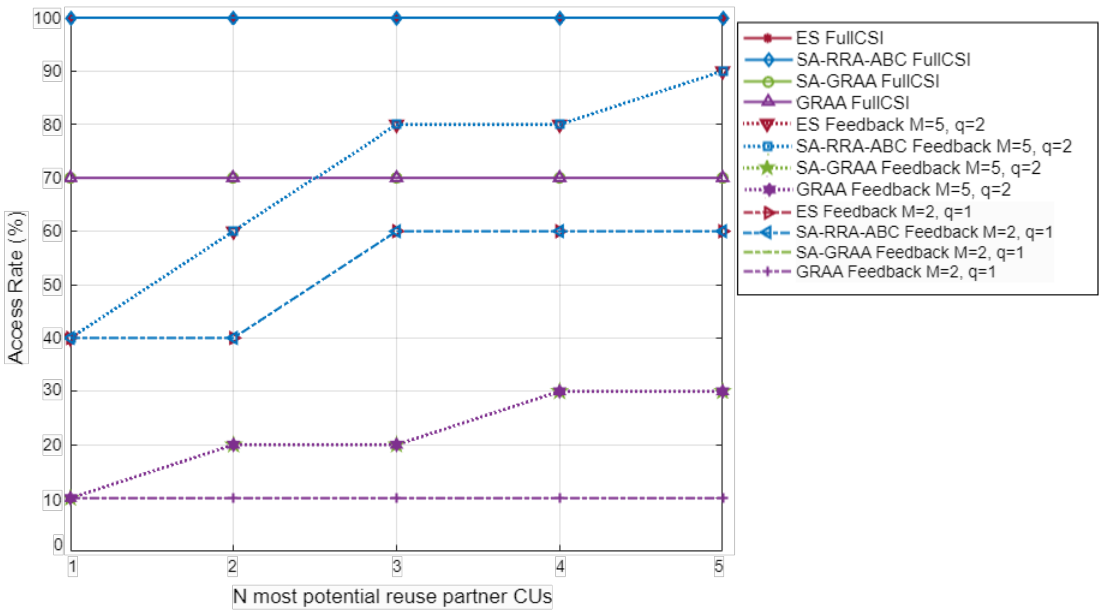
According to Figures 6.5(a) and 6.5(b), as N increases, more D2D communications are allowed in the system, thus, increasing throughput. When eNB receives information from CSI through several cellular channels, a better mapping of D2D pairs can be performed with the cellular channels of lowest mutual interference, and a larger number of D2D pairs can be allocated on the same channel.

Figure 6.6 shows a comparison of the algorithms with varying feedback on the most promising D2D pairs for sharing the same channel. The performance of SA-RRA-ABC was similar to that of ES and the meta-heuristic could find the optimal solution. According to Figure 6.6(a), the gain in throughput is almost constant for small values of $M < 5$, i.e., eNB neither has enough information on the interference between D2D pairs, nor admits more than one D2D pair to reuse the same channel. Therefore, the access rate is constant (Figure 6.6(b)) and the number of D2D pairs admitted in the network is lower than or equal to the number of available channels. As eNB receives information on the interference between D2D pairs, a more efficient resource allocation can be performed and more than one D2D pair can use the same channel simultaneously, thus, increasing the system throughput.

Figure 6.7 displays a comparison of the algorithms with varying control bits sent with information on CSI. According to Figure 6.7(b), an increase in q does not significantly affect the rate of access of D2D pairs to the system. However, in relation to throughput, it increases granularity and provides eNB with feedback closer to the real value of the channel state of each D2D communication. As shown in Figure 6.7(a), the throughput significantly increases with the increase in q, thus, promoting a more efficient allocation of resources.

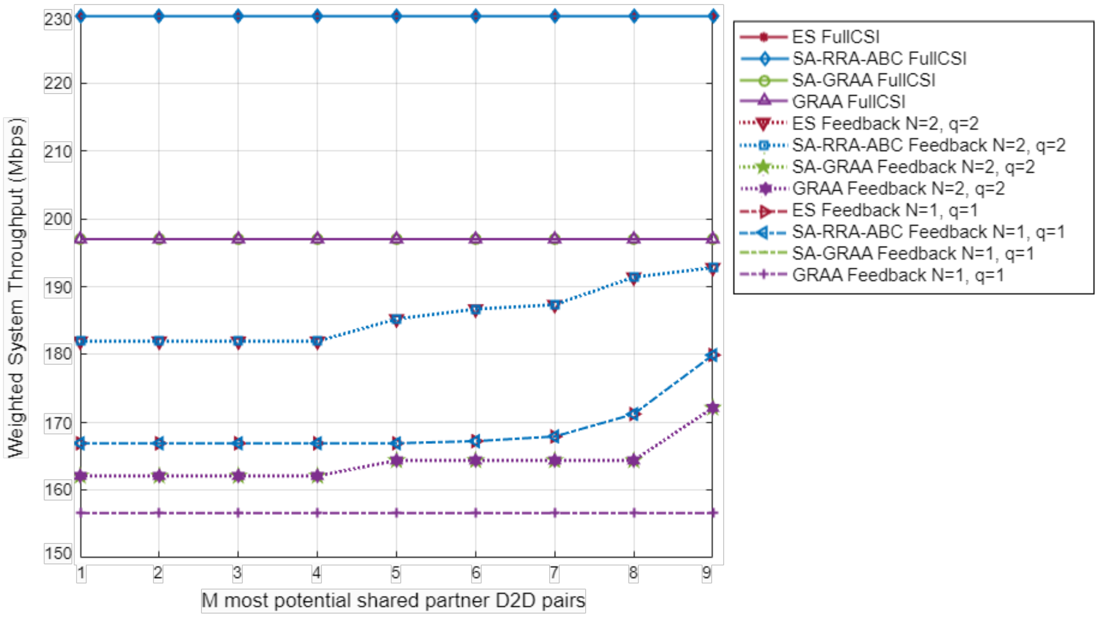


(a) R_T - Weighted System Throughput (Mbps)

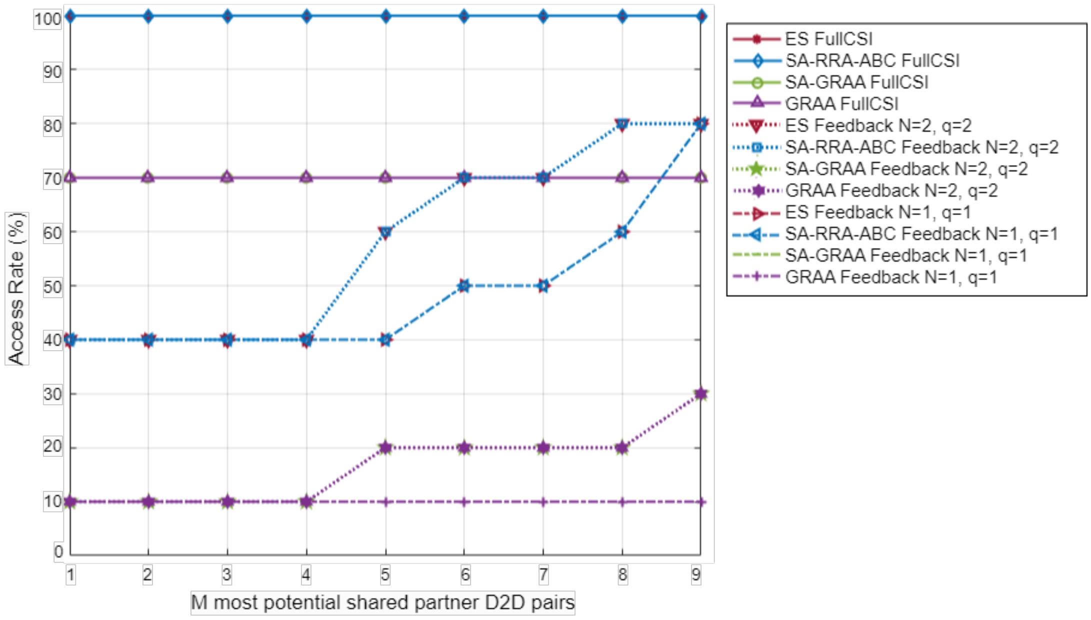


(b) Access Rate

Figure 6.5: Comparison of Algorithms varying the N feedback parameters ($u = 5$ and $d = 10$).

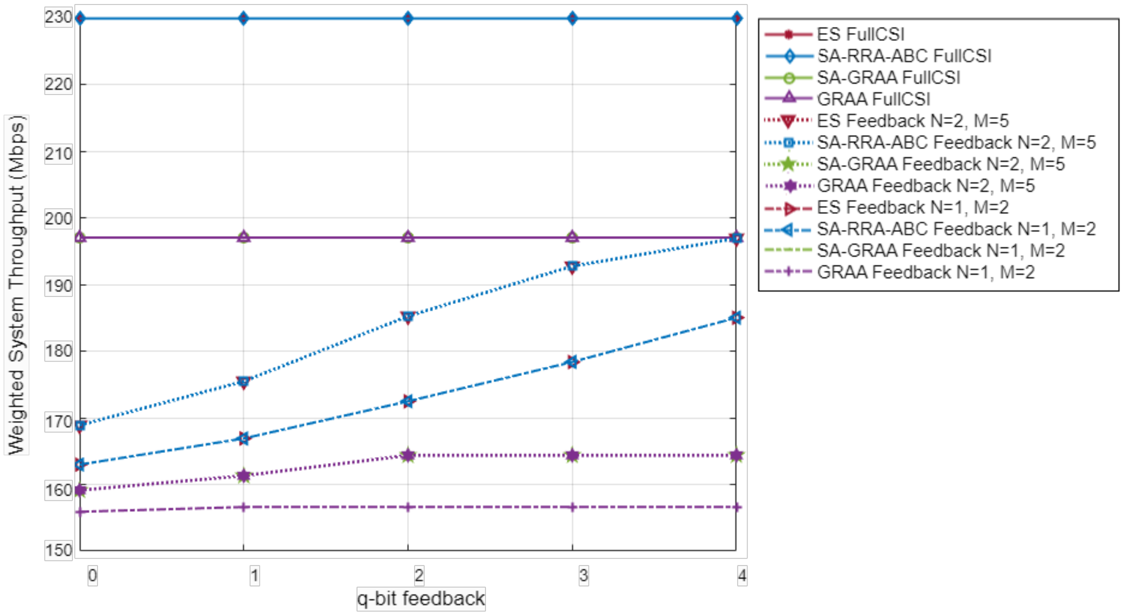


(a) R_T - Weighted System Throughput (Mbps)

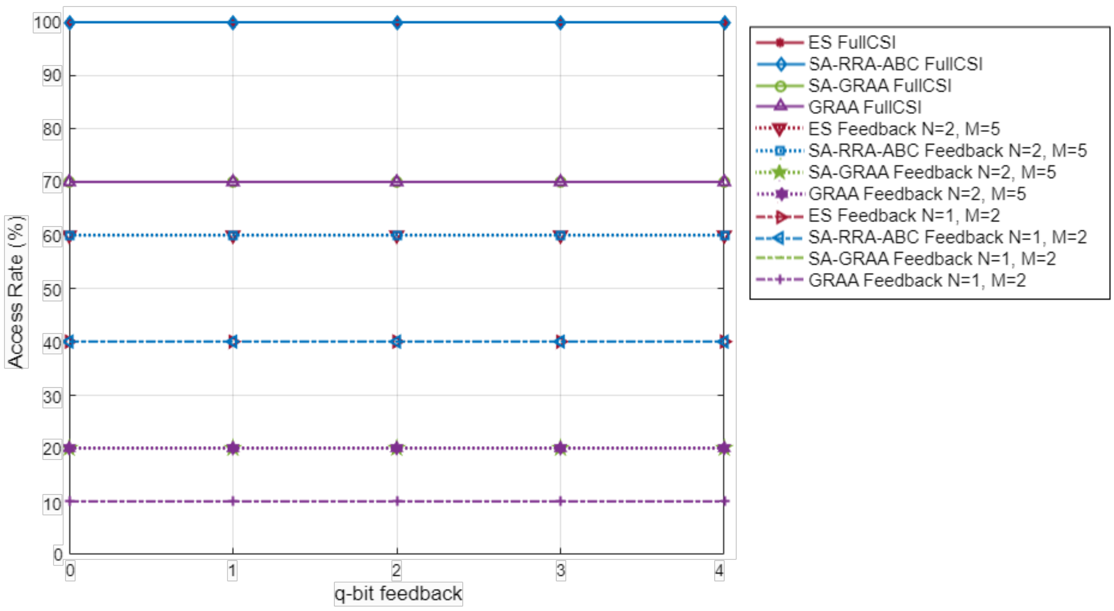


(b) Access Rate

Figure 6.6: Comparison of Algorithms varying the M feedback parameters ($u = 5$ and $d = 10$).



(a) R_T - Weighted System Throughput (Mbps)



(b) Access Rate

Figure 6.7: Comparison of Algorithms varying the q feedback parameters ($u = 5$ and $d = 10$).

A more realistic scenario with $u = 10$ subchannels and $d = 20$ D2D pairs is displayed in Figure 6.8, and trends are similar to those in Figure 6.5. Both weighted system throughput and access rate increase as eNB receives feedback from more potential reuse partners. SA-RRA-ABC showed better performance in the allocation of resources, maximizing the throughput of the system. As shown in Figure 6.8(a), for $N \geq 7$, the throughput gain is not significant and remains practically constant; therefore, eNB does not need information from CSI considering all potential reuse cellular partners for performing an efficient resource allocation.

Figure 6.9 displays the influence of increasing M in the most realistic scenario. According to Figures 6.9(a) and 6.9(b), more information on CSI is sent to eNB considering a higher number of potential shared D2D pairs, and a better mapping of number of D2D allocated in the same channel can be performed. The increase occurs because a D2D pair can be present in feasible D2D sets of more subchannels as M increases, implying its better chance of being assigned to a subchannel with high throughput. In the scenario under evaluation, for example, with $N = 7$, $M = 18$, and $q = 4$, the access rate achieved by SA-RRA-ABC was very close to that of the full CSI model, which represents the upper limit (Figure 6.9(b)). Therefore, a complete CSI is not necessary for the achievement of an efficient allocation, which reduces feedback overhead.

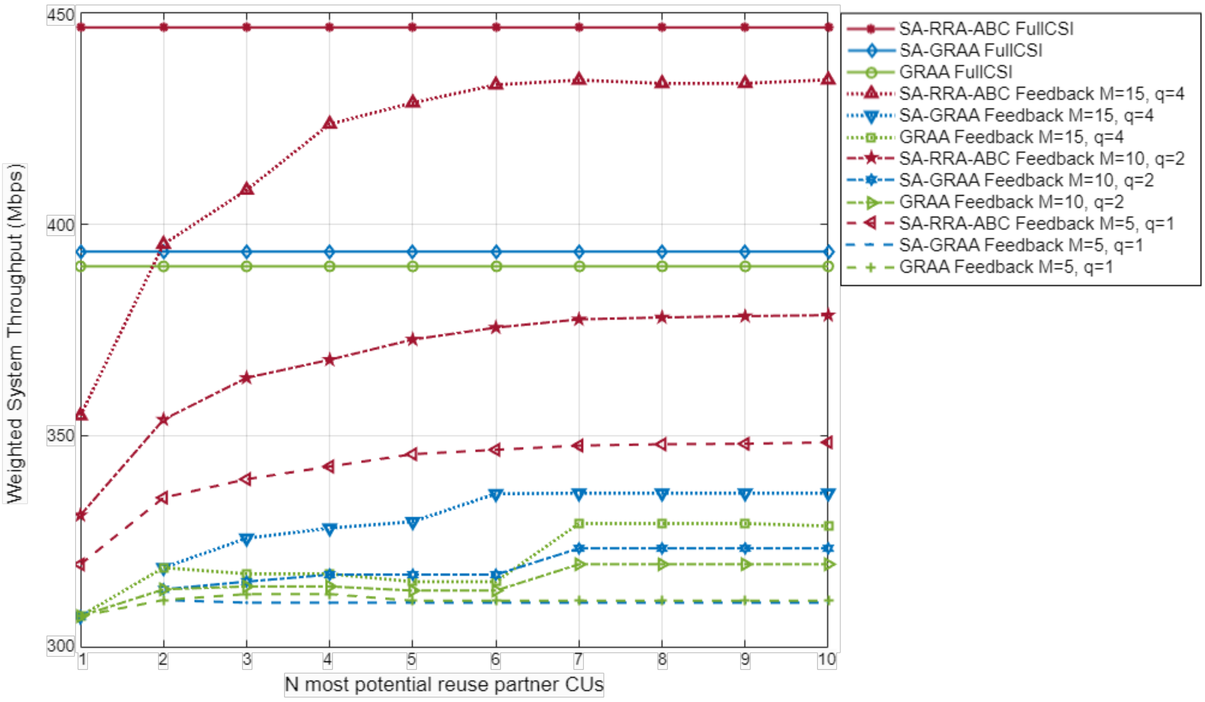
The trends in Figure 6.10 are similar to those in Figure 6.7, where the gain in granularity exerts no significant impact on the access rate. However, the weighted system throughput increases as q increases due to a better resolution of the feedback. As shown in Figure 6.10(b), an increase in the feedback resolution can decrease the access rate, since SA-RRA-ABC prioritizes D2D pairs with higher intensity in their social relationship. Therefore, a smaller number of D2D pairs admitted to the network may also represent an increase in throughput.

According to Figures 6.5, 6.6, 6.7, 6.8 and 6.9, SA-RRA-ABC achieved higher weighted system throughput and a higher number of D2D pairs was admitted in the network, thus, enabling a better use of the radio spectrum. SA-RRA-ABC maximized the weighted system throughput and used radio resources more efficiently. As a result, a larger number of pairs D2D reused the RBs assigned to CUE, which shows better performance of the meta-heuristic in solving the social-aware RRA problem.

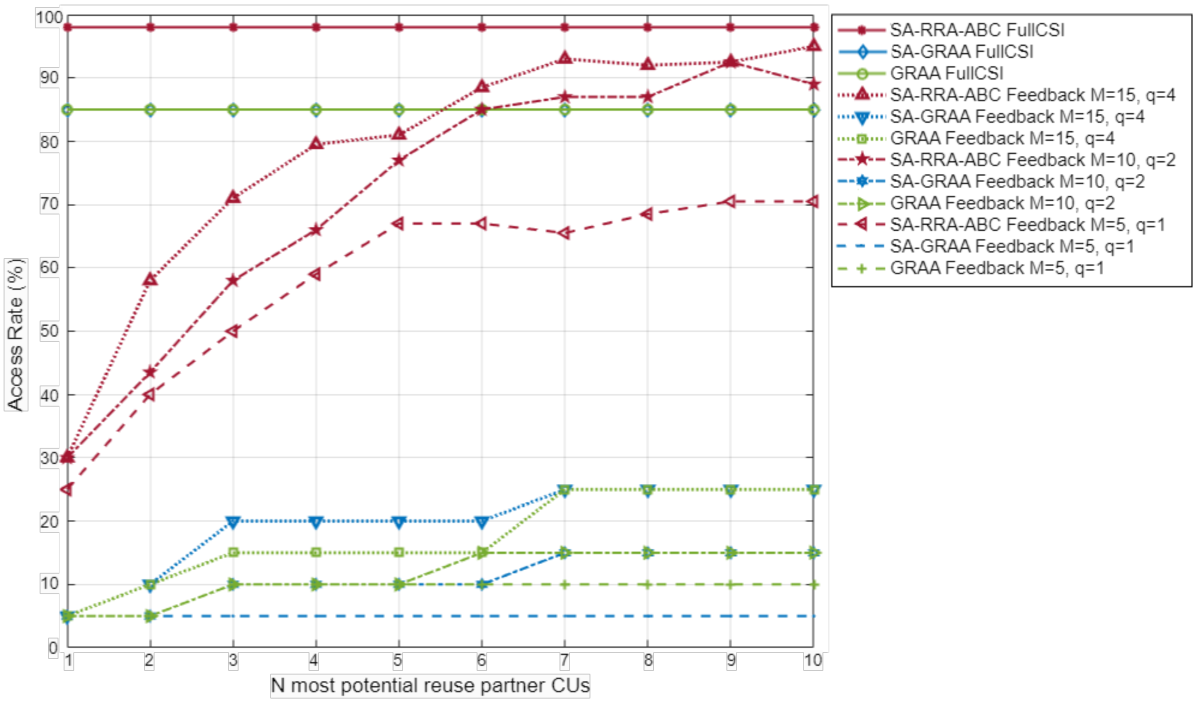
Figures 6.11, 6.12, and 6.13 show the result of increasing each parameter of the feedback model in the signaling overhead. In Figure 6.11, the slope of the lines is not accentuated, which shows N has a marginal contribution to the overhead increase; however, an increase in M and q significantly increases the overhead, and a two-fold increase in q is very representative.

Figure 6.12 displays a similar trend, i.e., and increase in M does not contribute to overhead significantly. However, feedback overhead grows linearly with the number of users. According to Figure 6.13, q is the parameter of major contribution to overhead and the most significant in throughput for the definition of feedback resolution.

The feedback overhead increases linearly with increasing N, M, and q parameters. Therefore, the feedback parameters must be defined in the design of the system according to their cost/benefit ratio.

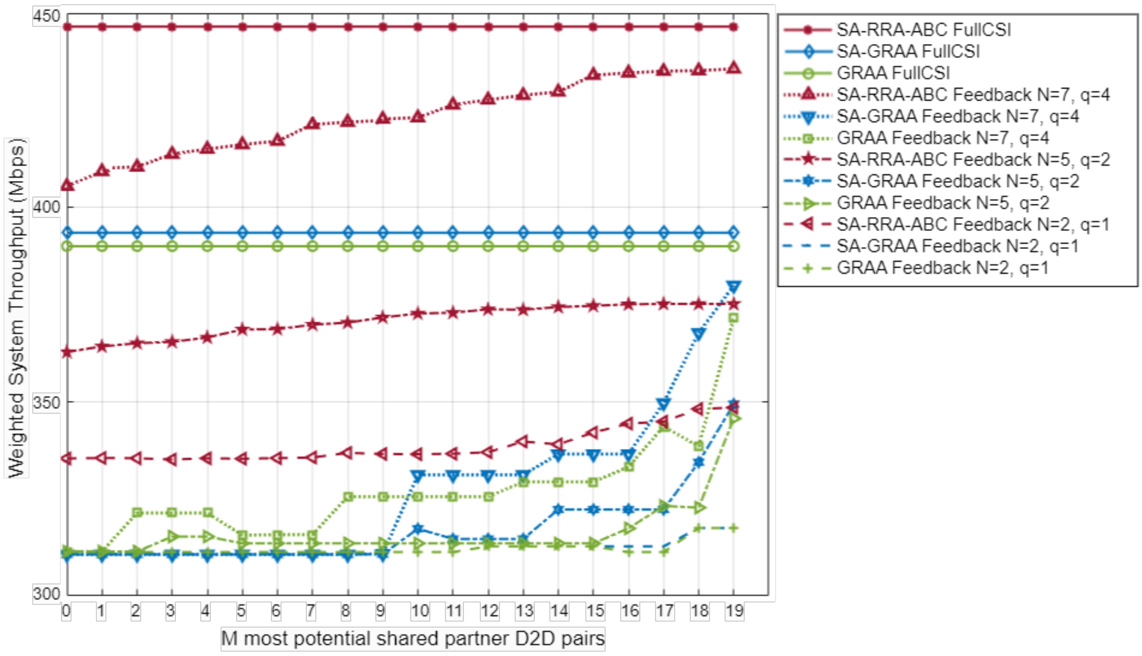


(a) R_T - Weighted System Throughput (Mbps)

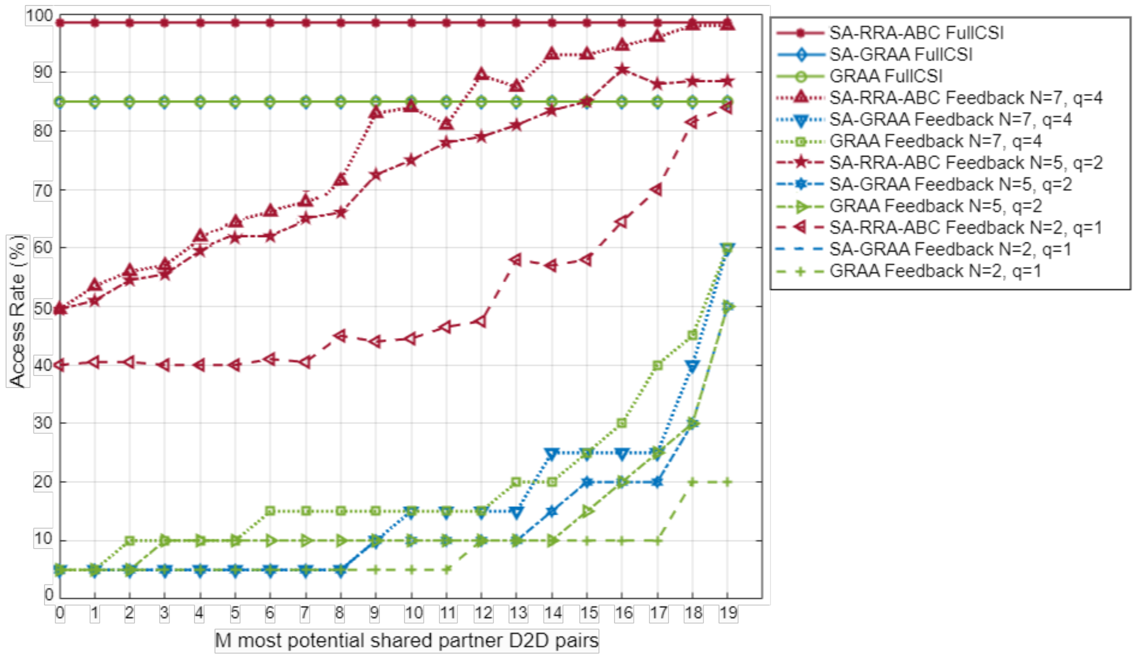


(b) Access Rate

Figure 6.8: Comparison of Algorithms varying the N feedback parameters ($u = 10$ and $d = 20$).

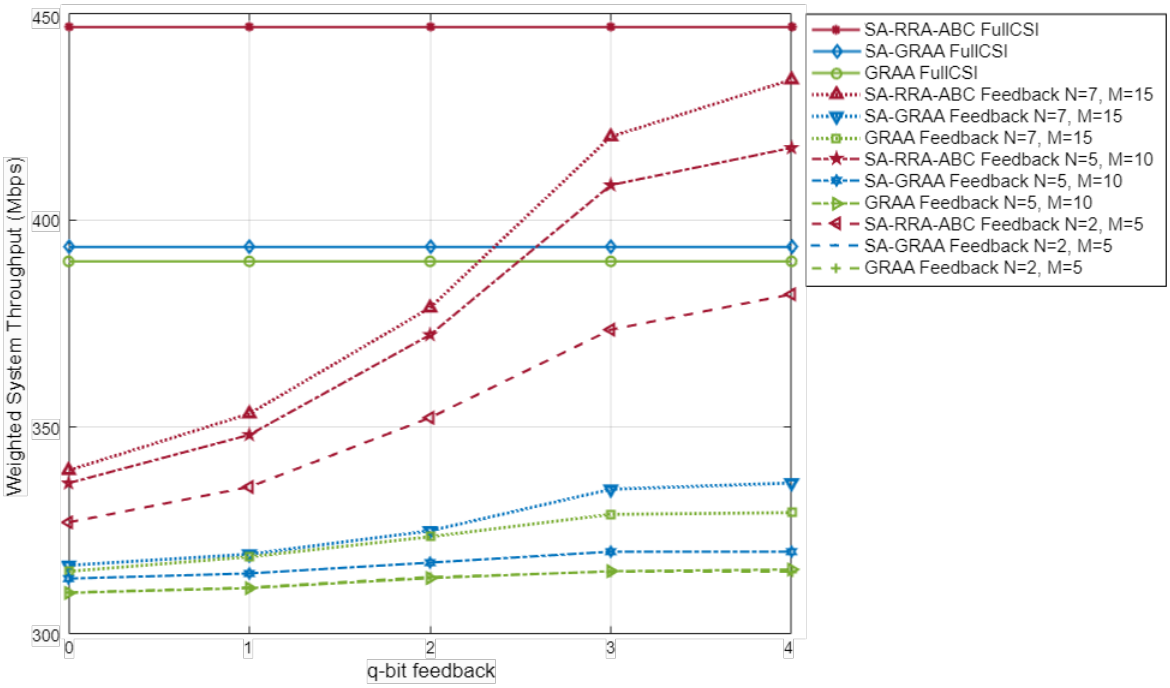


(a) R_T - Weighted System Throughput (Mbps)

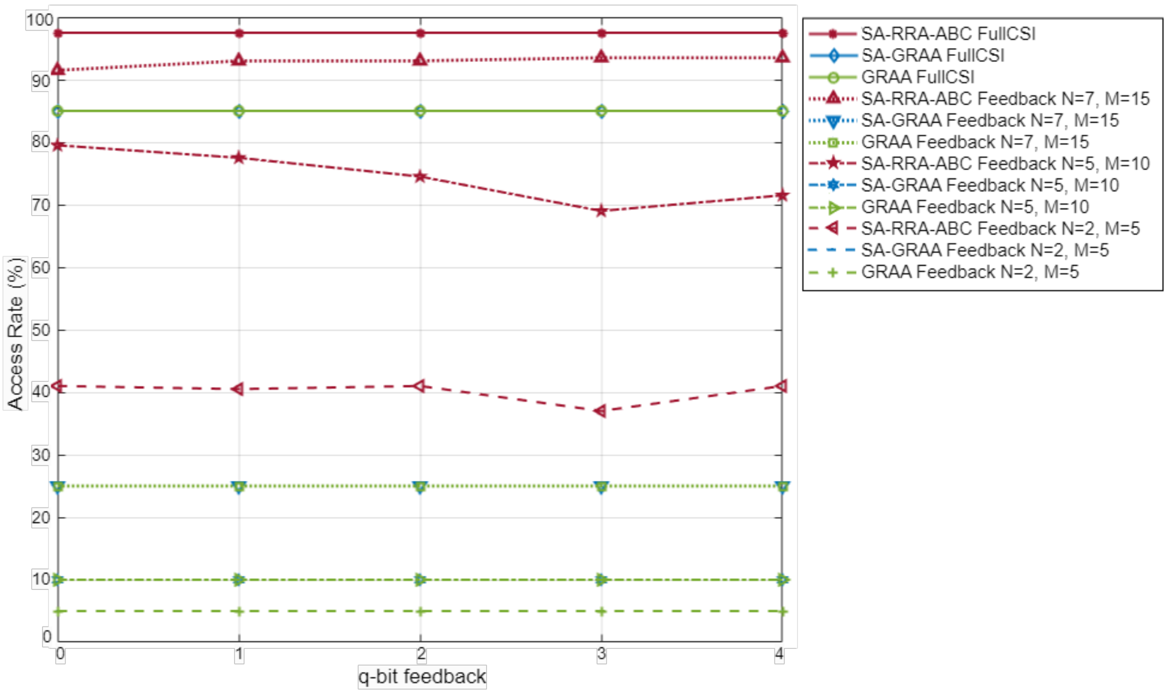


(b) Access Rate

Figure 6.9: Comparison of Algorithms varying the M feedback parameters ($u = 10$ and $d = 20$).

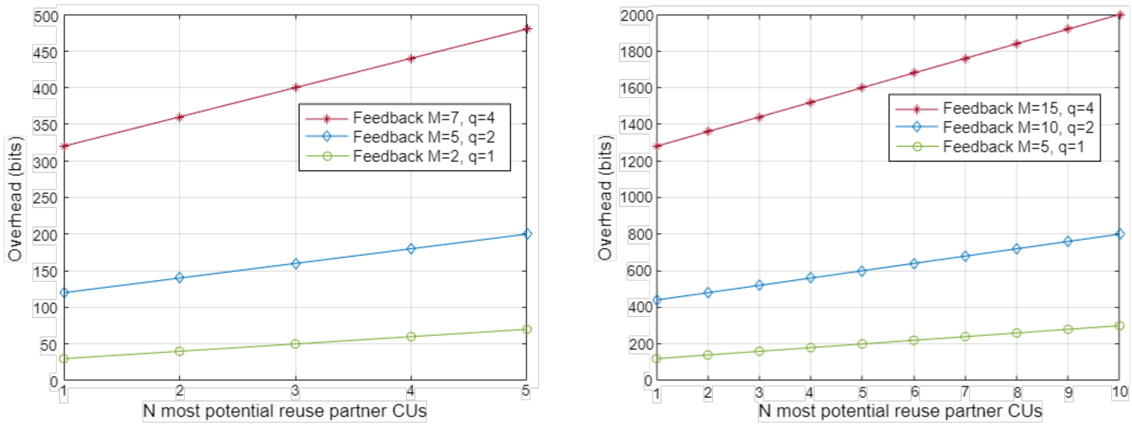


(a) R_T - Weighted System Throughput (Mbps)



(b) Access Rate

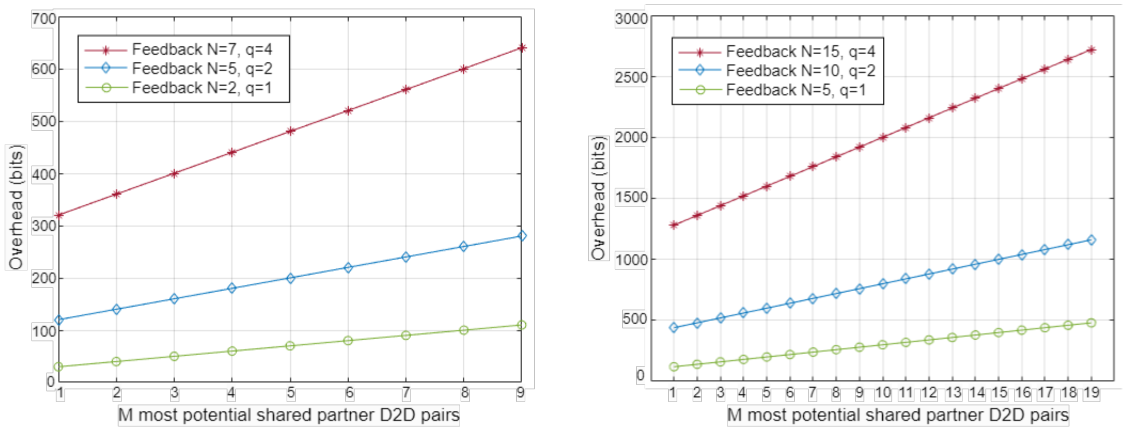
Figure 6.10: Comparison of Algorithms varying the q feedback parameters ($u = 10$ and $d = 20$).



(a) scenario $u = 5$ and $d = 10$.

(b) scenario $u = 10$ and $d = 20$.

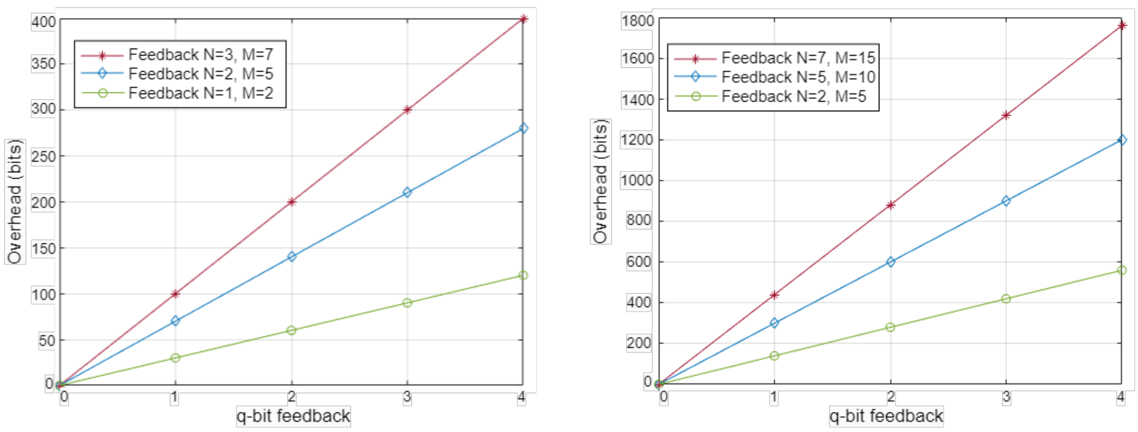
Figure 6.11: Feedback overhead varying the N feedback parameters.



(a) scenario $u = 5$ and $d = 10$.

(b) scenario $u = 10$ and $d = 20$.

Figure 6.12: Feedback overhead varying the M feedback parameters.



(a) scenario $u = 5$ and $d = 10$.

(b) scenario $u = 10$ and $d = 20$.

Figure 6.13: Feedback overhead varying the q feedback parameters.

Table 6.8: Simulation Parameters

Parameter	Value
system bandwidth - W	1 MHz
Number of CUEs and DUE pairs	{0,2,4} CUEs, 8 DUE pairs
N and N_u	4 RBs, {4,2,0} RBs
Maximum transmission power CUE and DUE	21 dBm, 10dBm
eNB Coverage Radio	500 m
Distance between D2D transmitter and receiver	Uniformly distributed among [0,20]m
Path loss model for cellular link	$128.1 + 37.6 \log_{10}(d[Km])$
Path loss model for D2D pairs	$148 + 40 \log_{10}(d[Km])$
Antenna gain	14 dBi for eNB; 0 dBi for UE
single-sided noise power spectral density - N_0	-174 dBm/Hz
SINR^t	10 dB
number of iterations	75

6.3 SECOND RRA OPTIMIZATION PROBLEM

A cellular network was considered for establishing D2D communications with URLLC services, with $\epsilon = 10^{-5}$ and $m = 500$ symbols (Table 6.8 shows the defined parameters).

The simulation environment involved programs written in C# for the definition of the values of ABC, ACS-EM, and PSO meta-heuristics. The parameters for ABC were $SN = 15$, $limit = 5$, and $NL_{size} = 5$. ACS-EM was applied with the following ones: $ant\_quantity = 15$; $q_0 = 0.2$; $r_0 = 0.4$; $z_0 = 0.6$; $\alpha = 0.1$; $\beta = 0.9$; $\tau_0 = 0.05$, and $\delta = 0.1$, and PSO was applied with $particle\_quantity = 15$; $w = 0.1$; $c1 = 2$, and $c2 = 2$.

The meta-heuristics were compared with a greedy heuristic and an ES algorithm that uses the Backtracking technique to validate its efficiency in solving the problem in question.

Greedy heuristic is based on the proposal in [75], which considers a sequential resource allocation scheme running at the eNB; it defines the D2D pairs admission and allocates resources according to the sequential indexing of RB. Conflict graphs are updated whenever a D2D pair has been selected to use a resource block in RM, and tolerable interference values are evaluated by both eNB and other DUE admitted.

The following three scenarios were considered in the evaluation: (i) $N_u = 0$ and $u = 4$, where all RBs were allocated for one CUE and only the reuse mode was possible for D2D communications, (ii) $N_u = 2$ and $u = 2$, where the network has two CUE and two unused RBs, and (iii) $N_u = 4$ and $u = 0$, where the network has no CUE and 4 unused RBs. Unused RBs can be assigned to D2D communications in either dedicated or cellular communication modes. The eNB is located in the cell center and the UEs are uniformly distributed in the cell.

The following four sets of weights were considered for evaluations of their effects on the quality of solutions (α_1, α_2, α_3, and α_4 - equation 5.6):

- *Set 1*: $\alpha_1 = 0.5$, $\alpha_2 = 0.5$, $\alpha_3 = 0$, and $\alpha_4 = 0$, representing an extreme case whose aim is to maximize the network energy efficiency by maximizing the data rate;
- *Set 2*: $\alpha_1 = 0$, $\alpha_2 = 0$, $\alpha_3 = 0.5$, and $\alpha_4 = 0.5$, representing an extreme case whose aim is to maximize the network energy efficiency by minimizing the transmission power;
- *Set 3*: $\alpha_1 = 0.25$, $\alpha_2 = 0.25$, $\alpha_3 = 0.25$, and $\alpha_4 = 0.25$, whose aim is to balance the weights and maximize the network energy efficiency considering equal measure data rate and transmission power; and
- *Set 4*: $\alpha_1 = 0.3$, $\alpha_2 = 0.5$, $\alpha_3 = 0.1$, and $\alpha_4 = 0.1$, whose aim is to maximize the network energy efficiency by providing more weight to the maximization of the data rate - especially of theD2D communications – but still performing a power control.

The impact of the weights on different metrics was studied and the metrics considered were number of D2D admitted to the network, system throughput, power consumption, and network energy efficiency.

Figure 6.14 shows the number of D2D communications admitted in the network when the number of D2D pairs increases in the three scenarios. The four sets considered were evaluated, and *Set 2* and *Set 3* led to a smaller number of D2D communications. In the scenario where all RBs are used by cellular communications ($N_u = 0$), those sets (*2* and *3*) do not admitD2D communications, due to the values of weights α_3 and α_4. According to the results, the admission of D2D communications in the network negatively impacts power consumption, thus, decreasing the network energy efficiency. Therefore, they do not represent good solutions to be chosen - the higher the values of α_3 and α_4, the stronger the impact of power consumption on the value of the objective function. The previous result led the meta-heuristics to choose solutions with lower power consumption in the network, even if D2D communications were not established.

Figures 6.15 and 6.17 display the average throughput and average power consumption when the number of D2D communications, respectively, is varied. According to Figure 6.15, *Sets 1* and *4* achieved the highest network throughput values. The previous result was conditioned by the values of α_1 and α_2, which increase the importance of throughput in the objective function. As a result, larger values of α_1 and α_2 represent solutions in which the maximization of throughput exerts a greater impact; they are considered better solutions. However, as shown in Figures 6.16 and 6.17, they have lower values of network energy efficiency and higher power consumption in most scenarios, highlighting the tradeoff between throughput and power.

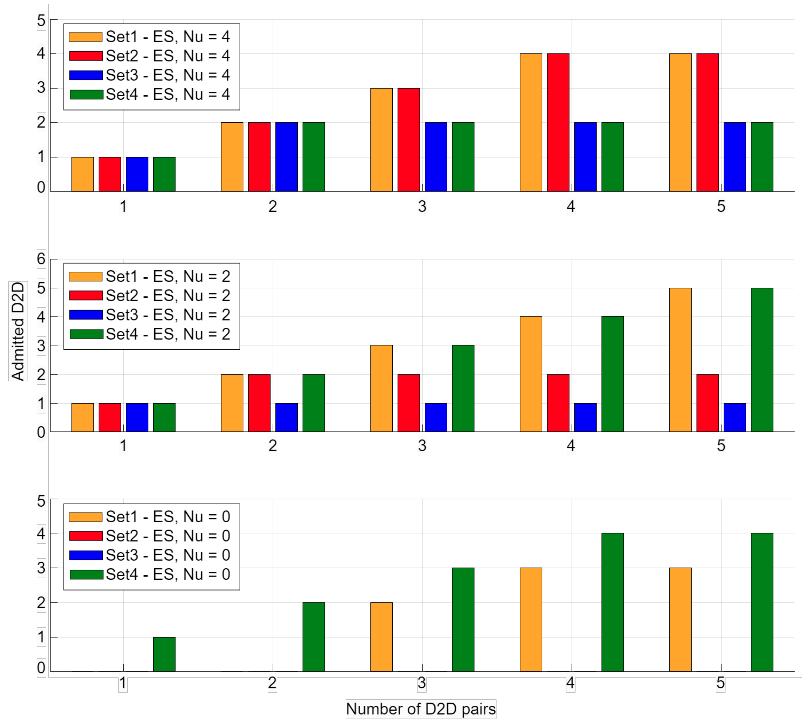


Figure 6.14: Number of admitted D2D pairs considering different weight sets.

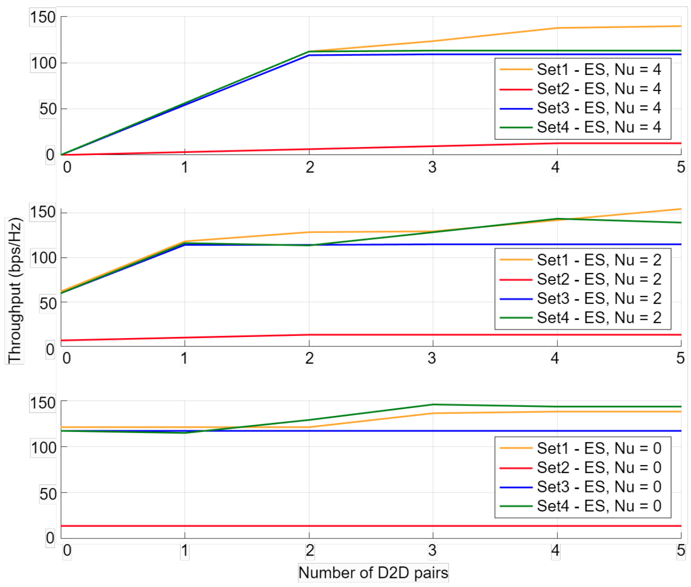


Figure 6.15: R_T - Throughput (bps/Hz) considering different weight sets.

Figure 6.16 shows the network energy efficiency for a varied number of D2D communications. According to Figures 6.16(a) and 6.16(b), *Sets 2* and *3* achieved the highest η_{EE} values. As explained elsewhere, in scenarios that require resource sharing, those solutions do not admit D2D communications in the network; therefore, *Sets 1* and *4* are more attractive, since they admit a larger number of D2D communications in all scenarios. As shown in Figure 6.16(b), higher energy efficiency is achieved when *Set 4* is used, in comparison to *Set 1*. The previous result shows the non adoption of power parameters in the objective function ($\alpha_3 = 0$ and $\alpha_4 = 0$) negatively impacts the energy efficiency of the network, despite a maximization of the throughput (Figure 6.15).

One of the aims of the present study is to admit the largest number of D2D communications in the network. Therefore, the parameter values must be chosen according to the tradeoff between throughput and power consumption and the need for admitting D2D communications. *Sets 2* and *3* are not good choices, since they do not admit D2D communications in resource sharing scenarios, whereas *Sets 1* and *4* represent better solutions, since they admit a higher number of D2D communications in all scenarios.

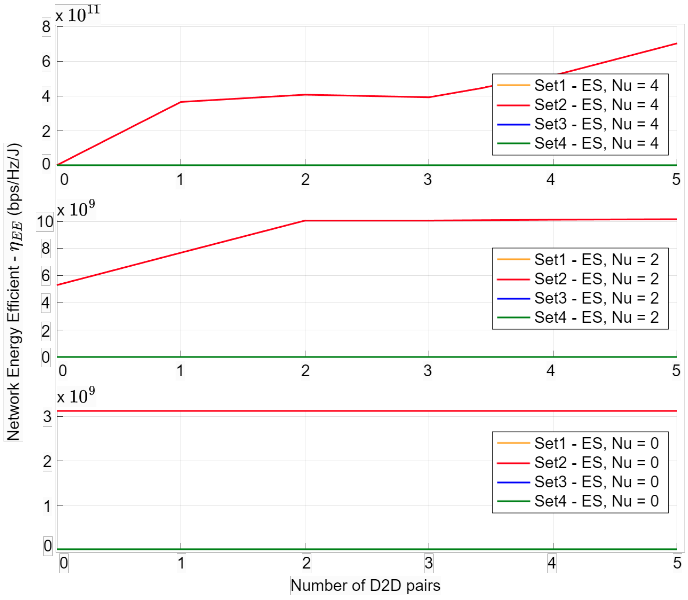
Set 4 is more appropriate than *Set 1*, since higher network energy efficiency is obtained (Figure 6.16(b)), thus, admitting D2D communications in the network with acceptable throughput values (Figure 6.15) and lower power consumption (Figure 6.17(b)) in most scenarios studied.

Figure 6.18 displays the objective function value (Equation 5.6) of the five algorithms studied when the D2D pair number was varied. Greedy heuristic shows the smallest value, therefore, its solutions are not good. ABC, ACS-EM, and PSO obtained better solutions to the RRA-URLLC problem, representing, in some cases, optimal or near optimal solutions. ABC achieved the same objective function value as ES in most scenarios evaluated, showing the best meta-heuristic for solving the problem.

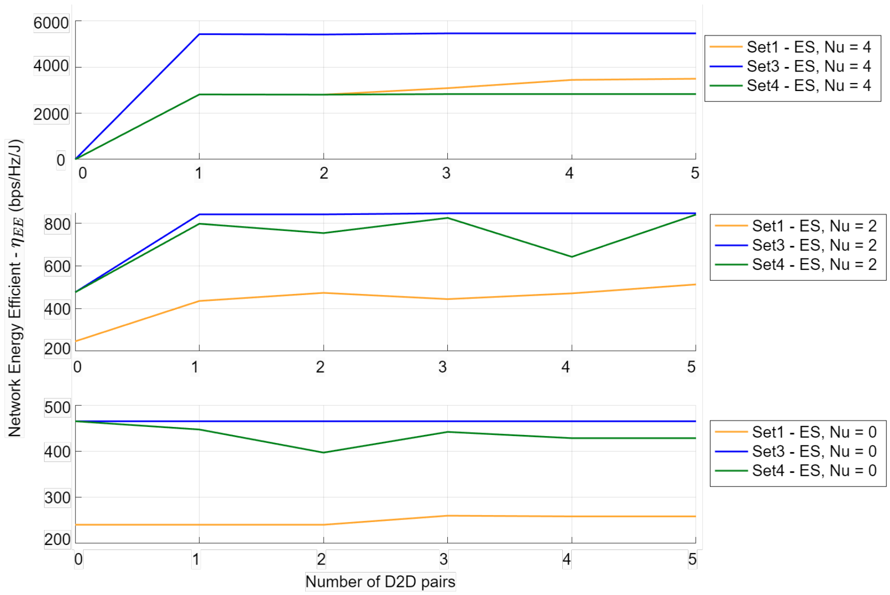
Figure 6.19 depicts the number of D2D communications admitted in the network when the number of D2D pairs increases in the three scenarios. In all cases, ABC and ACS-EM assigned a channel to each D2D pair similarly to the ES algorithm, showing its effectiveness in the three scenarios.

Figure 6.20 displays the number of D2D communications admitted in each communication mode when the D2D pair number increases for the three scenarios studied. Figure 6.20(a) shows scenario (i), where all RBs are used by cellular communications ($N_u = 0$) and all admitted D2D communications use RM mode, as long as the QoS parameters are respected. Figure 6.20(b) depicts scenario (ii), where half of the RBs are unused ($N_u = 2$) and the best solutions choose CM and RM modes, admitting all D2D pairs in the network. Figure 6.20(c) shows scenario (iii), where all RBs are unused ($N_u = 4$) and the best solutions choose CM mode, since they achieve higher throughput values by using two RBs. ABC showed the same communication mode choices of ES and its effectiveness.

Figure 6.21(a) shows an increase in energy efficiency as the number of D2D pairs increases in all scenarios, due to the contribution of each D2D communication to the system throughput (Fig-

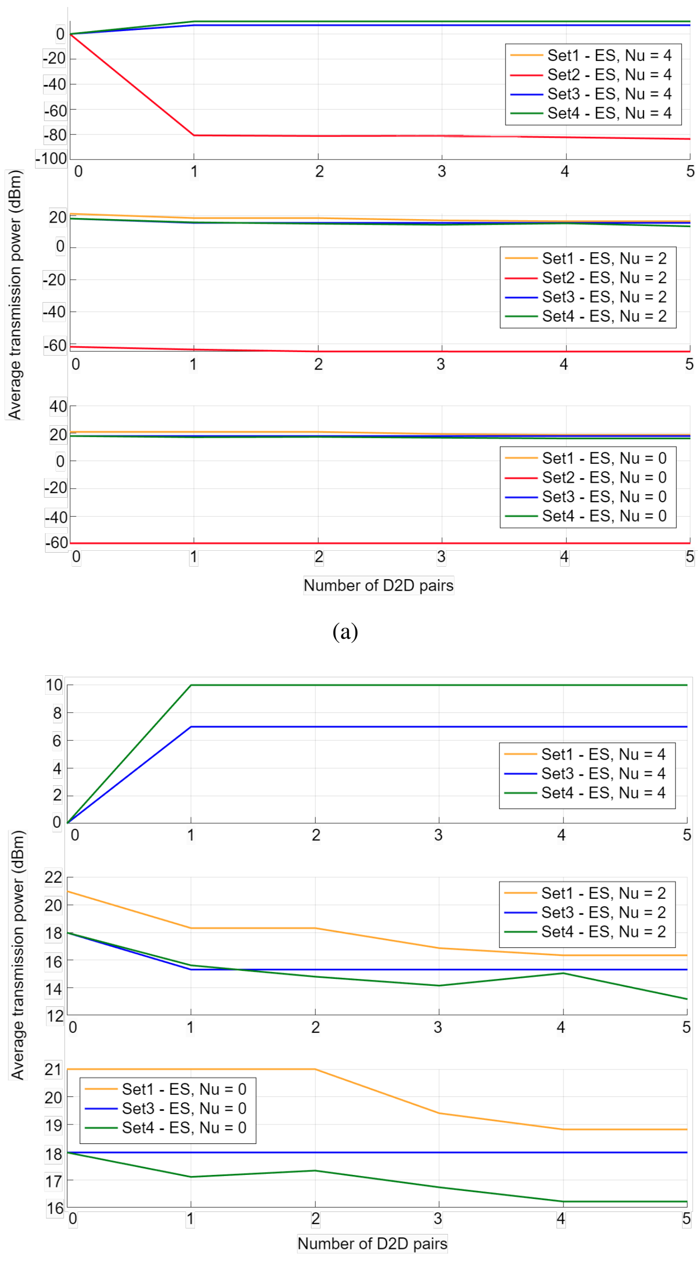


(a)



(b)

Figure 6.16: Comparison of the Network Energy Efficiency - η_{EE} considering different weight sets.



(a)

(b)

Figure 6.17: Comparison of the average transmission power (dBm) considering different weight sets.

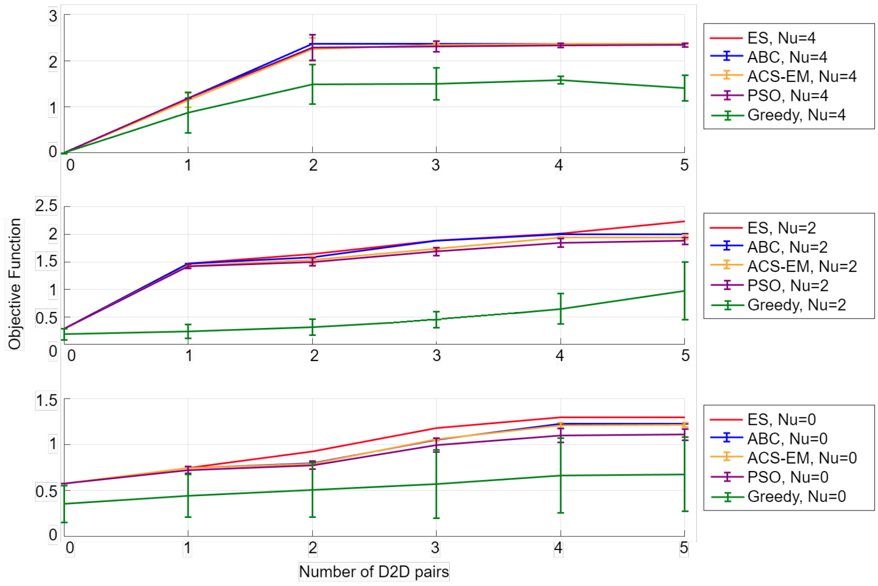


Figure 6.18: Comparison of the algorithms with a varied number of D2D pairs.

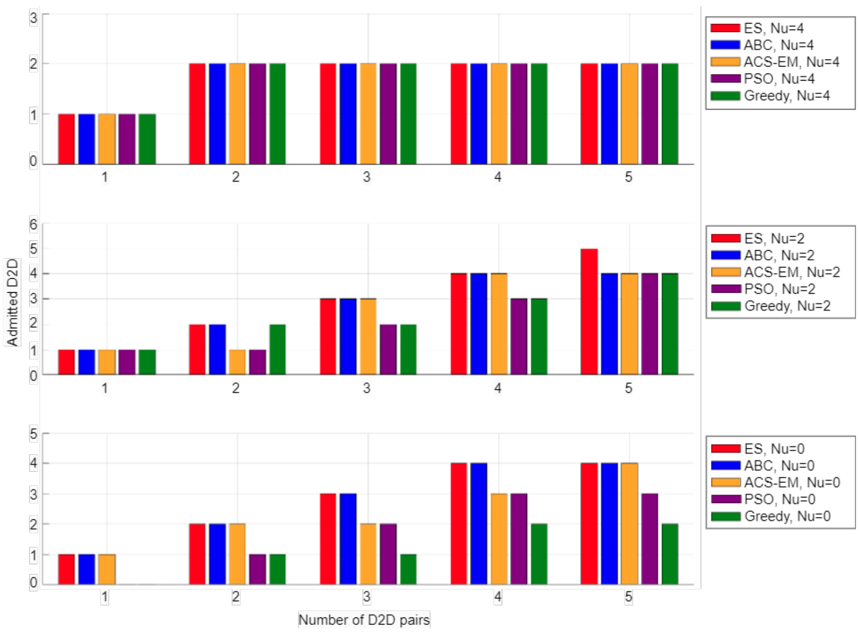
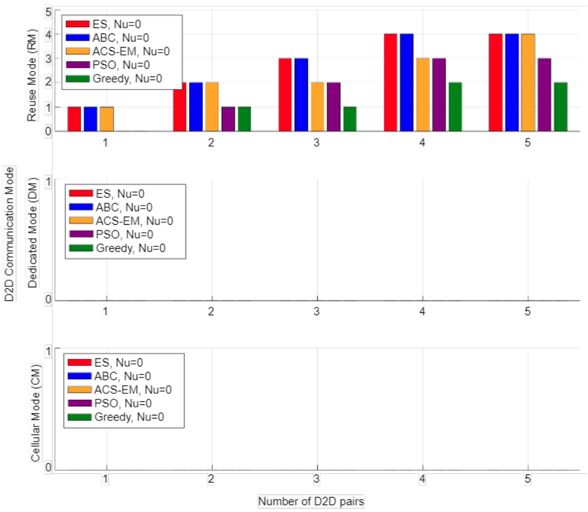
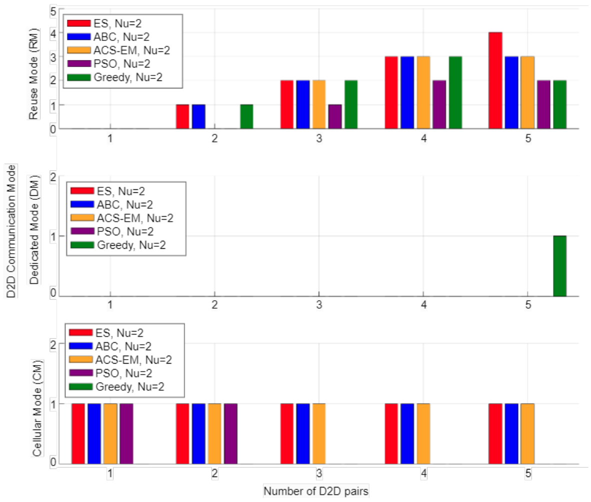


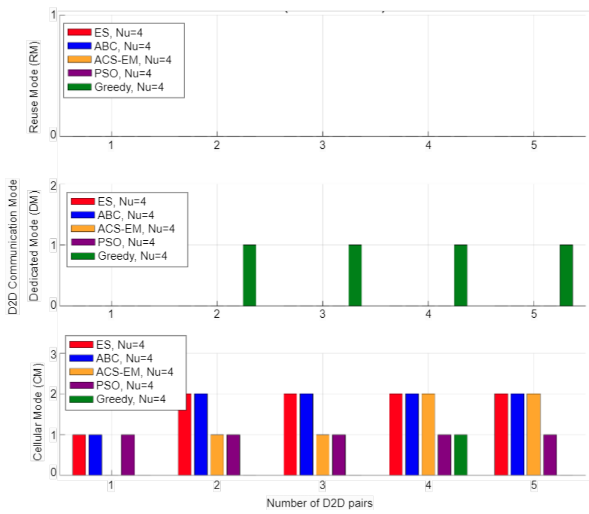
Figure 6.19: Number of admitted D2D pairs with a varied number of D2D pairs.



(a) No unused RB - $N_u = 0$

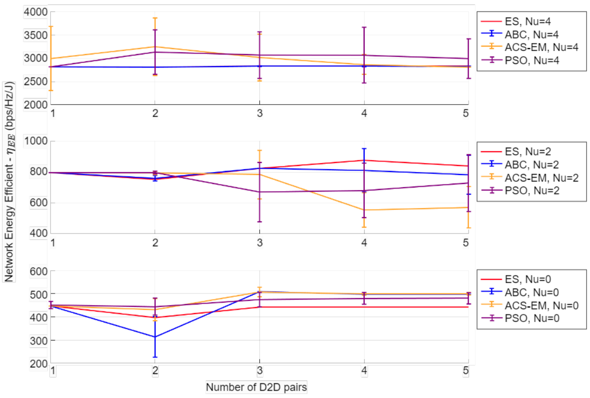


(b) Half of the unused RBs - $N_u = 2$

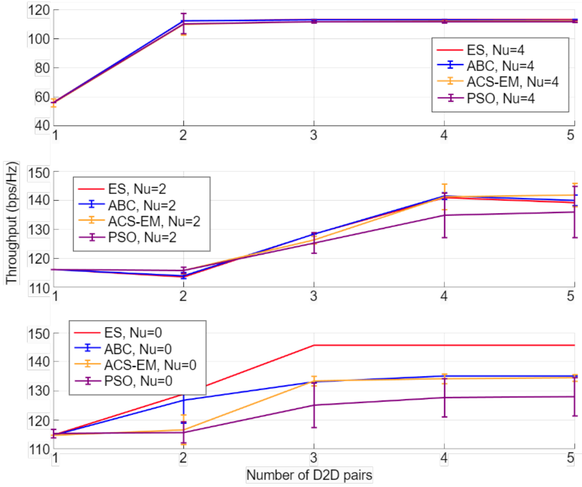


(c) All RBs unused (no cellular users) - $N_u = 4$

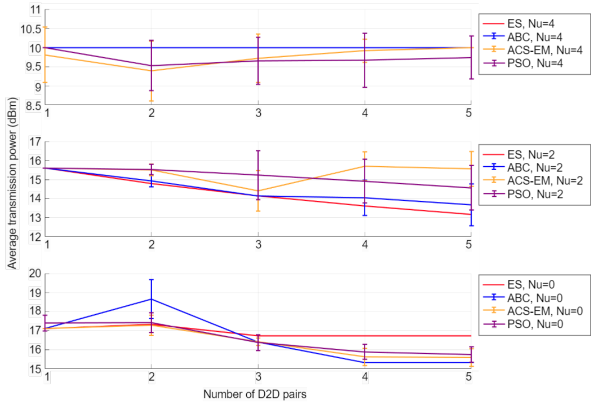
Figure 6.20: Comparison of the communication mode selection with a varied number of D2D pairs.



(a) Network Energy Efficiency - η_{EE}



(b) R_T - Throughput (bps/Hz)



(c) Average transmission power (dBm)

Figure 6.21: Comparison of the algorithms with a varied number of D2D pairs.

ure 6.21(b)) and the decrease in the average transmission power of the network (Figure 6.21(c)). However, in the scenario with $N_u = 4$, lower values of η_{EE} and the system throughput were obtained because of the co-channel interference generated by cellular and D2D communications in the reuse mode while sharing the same RB (Figures 6.21(a) and 6.21(b)). According to Figure 6.21(c), a successful communication requires transmission with higher power. Therefore, in the reuse mode and use of URLLC services, D2D communications increase the transmission power and decrease both throughput and energy efficiency in the system.

Figure 6.22(a) shows the throughput versus packet size (m) obtained by ABC and exhaustive search resource allocation schemes. First, the system throughput increases monotonically with increasing packet size and becomes linear in the region where the packet size is larger than 500 symbols, showing D2D communications with the use of URLLC services (small packet size < 500 symbols) negatively impact the system efficiency.

Figure 6.22(b) displays the impact of the decoding error probability on the system throughput - in scenarios with URLLC services, which require high reliability ($\epsilon < 10^{-5}$), the system throughput decreases. The previous results are associated with the maximum number of bits to be sent in a finite size packet and obtained basically as a calculated expression, thus, incurring a penalty on ergodic capacity. The penalty is inversely proportional to m and directly proportional to ϵ. Therefore, when m and ϵ become larger, the penalty added to the maximum achievable rate in the finite packet size regime is gradually reduced.

The computational complexities of the five algorithms were analyzed (more details are shown in Appendix B) and are expressed as

$$
\begin{aligned}
\textit{Greedy} => \quad & \mathcal{O}\Big(ud + d + n\Big), \\
\textit{ABC} => \quad & \mathcal{O}\Big(nro\_it * SN * [SN + u + d + n]\Big), \\
\textit{PSO} => \quad & \mathcal{O}\Big(nro\_it * particle\_quantity * ((u + d) * n' + n))\Big), \\
\textit{ACS-EM} => \quad & \mathcal{O}\Big(nro\_it * ant\_quantity * (n')^2 (u + d)^2)\Big), \\
\textit{ES} => \quad & \mathcal{O}\Big(u^n * d^{n'}\Big).
\end{aligned}
$$

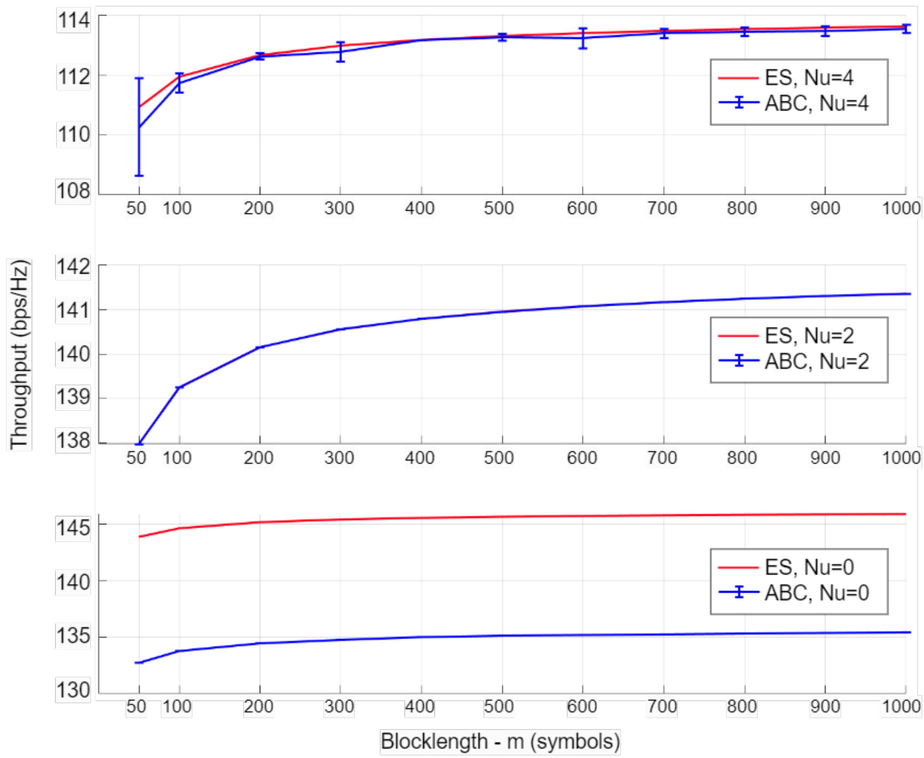
According to the computational complexity of ES, the computational effort increases exponentially with respect to the amount of RBs, CUEs, and D2D communications, making such a solution infeasible for sufficiently large u, d, and n. On the other hand, the complexity of Greedy, ABC, ACS-EM, and PSO is a polynomial function that provides good solutions with lower computational efforts, which makes the meta-heuristics good choices in scenarios with many D2D communications.

The analysis revealed the computational complexity of Greedy is lowest and those of ABC and PSO are lower than that of ACS-EM. However, ABC showed better performance in solving the problem, with characteristics that enable a more efficient exploration of the search space, thus, avoiding sub-optimal solutions.

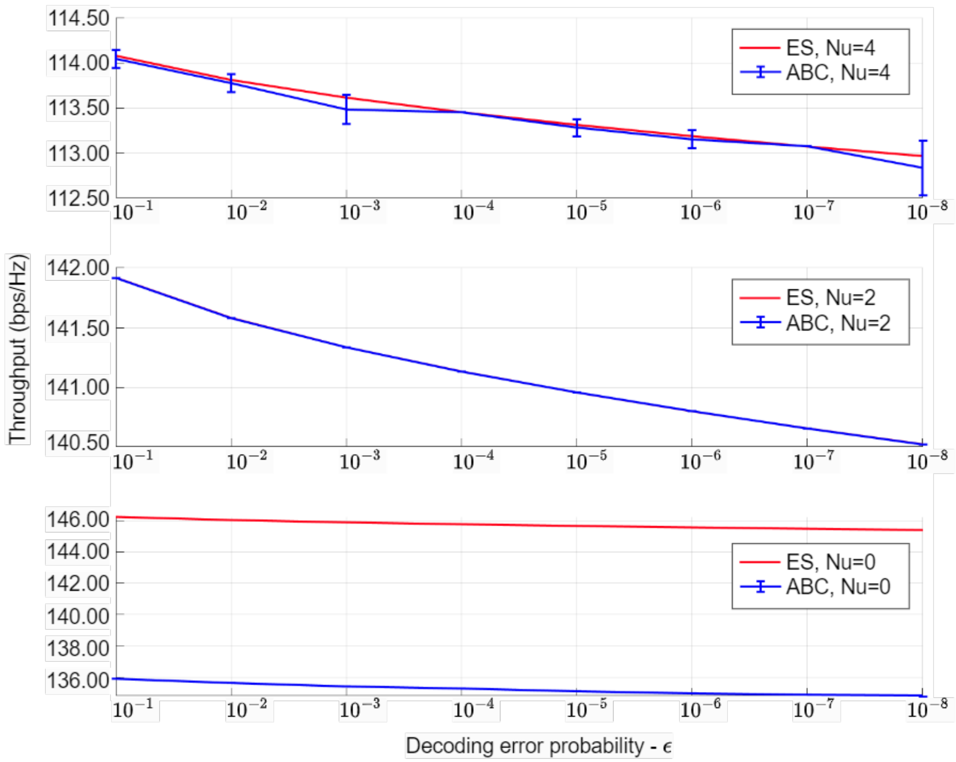
Finally, the convergence curves in Figure 6.23 show how the meta-heuristics reduce Mean Squared Error (MSE) values throughout iterations and improve the best solution. In a small number of iterations, ABC and ACS-EM reduce the MSE value efficiently. PSO does not reduce the MSE value with the same efficiency, and in the scenario with $N_u = 0$, it does not reach the desired value (MSE = 0), with solutions farthest from the optimal one. In this study, the maximum iteration number was 200.

6.4 FINAL CONSIDERATIONS

This chapter presented an evaluation of the proposed solutions to the CPP and RRA problems. Simulation results for the CPP problem were shown considering social-unaware and social-aware scenarios. The results show that when more than one controller is used, its best location is in the access network, at eNBs, and thus, near D2D communications, since eNBs represent the first hop, which reduces both propagation latency and total response time. CPP-ABC and CPP-ACS-



(a) Impact of blocklength on system throughput



(b) Impact of decoding error probability on system throughput

Figure 6.22: Comparison of algorithms varying m and ϵ parameters associated with URLLC services.

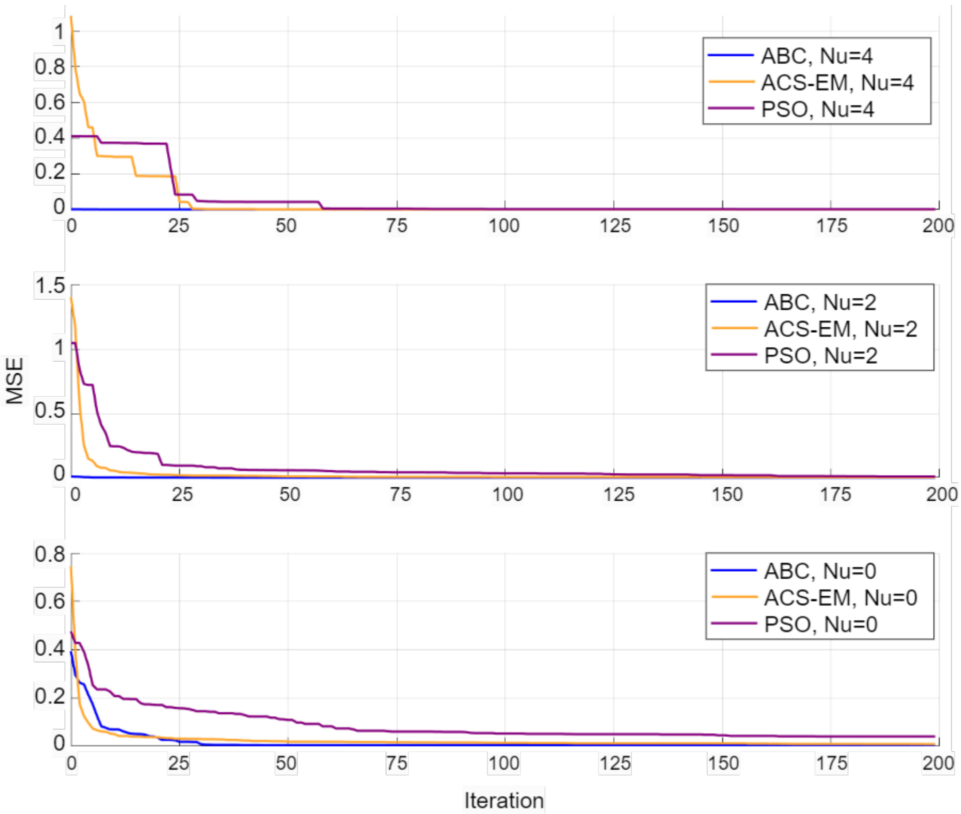


Figure 6.23: MSE convergence curves of the meta-heuristics.

EM were compared with CPP Ant Colony System (CPP-ACS) and Particle Swarm Optimization (CPP-PSO). The analysis revealed that the computational complexity of the CPP-ABC and CPP-PSO meta-heuristics is lower than that of the CPP-ACS-EM and CPP-ACS meta-heuristics. However, CPP-ABC and CPP-ACS-EM showed better performance in solving the problem according to statistical tests.

Regarding the RRA problem, two approaches were considered. The first aims to maximize the weighted system throughput considering the social relationship among users and a partial CSI model with limited feedback, while the second aims to maximize the network energy efficiency considering URLLC services. In evaluating the first proposed approach for solving the RRA problem, two metrics were used: D2D access rate and system weighted throughput. The results were presented in graphs varying the N and M parameters of the selection model used in the proposed feedback scheme. The SA-RRA-ABC meta-heuristic was compared with an exhaustive search algorithm, the GRAA and the SA-GRAA. According to the results obtained the SA-RRA-ABC achieved a higher weighted throughput of the system and a larger number of D2D pairs were admitted to the network, thus, allowing a better utilization of the radio spectrum. Also the feedback overhead was evaluated, where the results showed that it increases linearly with the increment of the N, M and q parameters.

In the last section the results of the second proposed approach to solving the RRA problem were presented. Three scenarios were considered in the evaluation and the RRA-URLLC-ABC, RRA-URLLC-ACS-EM and RRA-URLLC-PSO meta-heuristics were compared with an exhaustive search algorithm and a Greedy algorithm. First, the impact of four sets of objective function weights on the metrics, number of D2D admitted to the network, system throughput, power consumption, and network energy efficiency was studied. The results show the non adoption of power parameters in the objective function negatively impact network energy efficiency, despite a maximization of throughput. Therefore, the weights should be chosen according to the tradeoff between throughput and power consumption.

The impact of blocklength and decoding error probability on system throughput, in scenarios with URLLC services, was also evaluated. The results show that D2D communications using URLLC services have a negative impact on system efficiency. Additionally, the meta-heuristics were compared and the results showed that the computational complexity of Greedy is lower and those of RRA-URLLC-ABC and RRA-URLLC-PSO are lower than those of RRA-URLLC-ACS-EM. However, RRA-URLLC-ABC showed better performance in solving the problem.

Finally we can observe that the ABC meta-heuristic showed better results in solving the two problems CPP and RRA, which is due to a more efficient exploration of the search space, thus, avoiding sub-optimal solutions, while presenting reasonable computational complexity.

7 CONCLUSIONS AND FUTURE WORK

This study has addressed, initially, bioinspired-based optimization meta-heuristics for the controller placement problem in an LTE-A architecture with SDN-based social-aware D2D communications management. Simulations enabled analyses of the effect of social relationships on the number of SDN controllers. Social relationships between users can optimize the number of active SDN controllers on the cellular network, and the proposed meta-heuristic can perform a balanced assignment of the eNBs to the controllers, thus, guaranteeing a response time lower than or equal to a t_{QoS} predefined. ABC provided acceptable and accurate results in the scenario evaluated, in comparison to a discrete version of PSO, and found the global minimum in all simulations.

Moreover, the controller placement problem was analyzed for an LTE-A architecture with SDN-based D2D communications management, and two models were used to evaluate the effect of D2D communications management on the number of SDN controllers used in a cellular network. An interference model was used to obtain the maximum number of D2D communications allowed in the network, considering radio resource allocation. A model based on the queueing theory was used for SDN controllers to obtain their average response time.

The solution of the CPP was then divided into two steps. In the first, two greedy heuristics are used to solve the resource allocation problem. In the second, we have proposed an approach for the artificial bee colony and an approach for the ant colony system with external memory meta-heuristics for the evaluation of CPP.

The objective was to minimize the response time considering the SDN controller average response time and propagation latency among the eNB and the controllers, with some interference constrains. $C\#$ simulations checked the effect of D2D communications on the number, location and assignment of SDN controllers. The ABC and ACS-EM yielded acceptable and accurate results in the different scenarios evaluated, in comparison to ACS and a discrete version of PSO meta-heuristics. However, the ABC presented less computational complexity, which makes it a more attractive solution in time sensitive scenarios.

On the other hand, two optimization problems related to radio resource allocation (RRA) in LTE-A and 5G networks have been addressed. The first considered the social-aware resource allocation problem for D2D communication underlaying fully loaded cellular system with uplink spectrum reuse. Information on both social and physical layers was exploited in the optimization problem. The social relationship between two users was modeled according to their common features and friendship. A SA-RRA-ABC developed maximized the weighted system throughput while guaranteeing the QoS requirements of both cellular and D2D links simultaneously. Towards decreasing the feedback overhead, a selected-NM q-feedback model was designed to allocate several D2D pairs in the same channel, thus, increasing the weighted system throughput. The meta-heuristic was validated through simulations and compared with two greedy heuristics, namely GRAA and SA-GRAA. Simulation results showed its better performance. The selected-NM q-feedback model proposed can achieve performance close to that of the full CSI model with lower overhead. Furthermore, SA-RRA-ABC enables the admission of a higher number of D2D pairs in the network, increasing the reuse factor of RBs and using the spectrum more efficiently.

The second problem introduced three approaches based on a bioinspired meta-heuristic, called RRA-URLLC-ABC, RRA-URLLC-ACS-EM, and RRA-URLLC-PSO for solving the radio resource allocation problem in a D2D-enabled cellular network targeting wireless industrial applications in factories of the future and using URLLC services. RRA-URLLC problem was modeled as an optimization problem towards maximizing the network transmission power efficiency and meeting the QoS constraints imposed by cellular and D2D communications. The schemes were compared with a greedy heuristic and an exhaustive search algorithm and the performance of ABC was similar to that of ES, achieving near optimal solutions. Simulations validated the approaches and the impact of D2D communications using URLLC services on the system throughput was analyzed.

Our results for CPP have been published in two articles - one entitled "*Alocação de controladores SDN para gerenciamento de comunicações D2D*" and published at the SBrT2019 symposium (the initial page of the manuscript is shown in Appendix C), and another published in the IEEE Access journal (it was entitled "SDN-Controller Placement for D2D Communications" and its initial page is attached to Appendix D).

Part of our results for RRA have been published in Computer Networks journal (Elsevier), in an article entitled "Social-aware spectrum sharing for D2D communication by artificial bee colony optimization", attached to Appendix E (initial page of the article). A second article, entitled "Resource allocation based on bio-inspired algorithms for D2D cellular networks with URLLC services", has been submitted to Special Issue in Recent Advances in the Design and Management of Reliable Communication Networks of the IEEE Transactions on Network and Service Management journal (its initial page is shown in Appendix F) and is under review. A third article, entitled "Energy-efficient optimization for URLLC services in D2D cellular networks", whose initial page and acceptance letter are attached to Appendix G, has been accepted for presentation to the event "2023 26th Conference on Innovation in Clouds, Internet and Networks and Workshops (ICIN)".

As future work, we intend to solve the CPP considering other metrics, such as reliability and distance between controllers, and use of network emulators (e.g., Mininet) to evaluate the effectiveness of the proposed system. The influence of different radio resource allocation methods on CPP and aspects related to traffic differentiation for multimedia applications will also be evaluated. The effect of users' mobility on D2D communications and its influence on SDN controllers will be considered.

The design of social-aware resource allocation problem for D2D communication by incorporating other meta-heuristics based on populations, such as AFS, BFO, Firefly Algorithm (FA) and Whale Optimization Algorithm (WOA), an evaluation of the dynamic radio resources allocation considering metrics, such as user mobility, and the implementation of social IoT solutions for

D2D communications are also aimed at.

Scenarios with D2D clusters, where bio-inspired meta-heuristics can be used for solving problems such as resource allocation and interference mitigation will be considered, and Machine Leaning (ML) solutions will be employed towards solving the resource allocation problem in wireless industrial networks.

A multi-objective version of ABC will also be considered in order to further study the two optimization problems (CPP and RRA). Other non-parametric statistical significance tests will be performed to compare the results of the meta-heuristics used to solve the problems (for example Friedman's test).

BIBLIOGRAPHIC REFERENCES

[1] Alireza Farshin and Saeed Sharifian. A chaotic grey wolf controller allocator for Software Defined Mobile Network (SDMN) for 5th generation of cloud-based cellular systems (5G). Computer Communications, 108:94 – 109, 2017. ISSN 0140-3664. doi: 10.1016/j.comcom.2017.05.003.

[2] Erik Dahlman, Gunnar Mildh, Stefan Parkvall, Janne Peisa, Joachim Sachs, and Yngve Selén. 5G radio access. Ericsson review, 6:2–7, 2014.

[3] E. Hossain and M. Hasan. 5G cellular: key enabling technologies and research challenges. IEEE Instrumentation Measurement Magazine, 18(3):11–21, June 2015. ISSN 1094-6969. doi: 10.1109/MIM.2015.7108393.

[4] W. H. Chin, Z. Fan, and R. Haines. Emerging technologies and research challenges for 5G wireless networks. IEEE Wireless Communications., 21(2):106–112, April 2014. ISSN 1536-1284. doi: 10.1109/MWC.2014.6812298.

[5] A. Asadi and V. Mancuso. WiFi Direct and LTE D2D in action. 2013 IFIP Wireless Days (WD), pages 1–8, Nov 2013. ISSN 2156-9711. doi: 10.1109/WD.2013.6686520.

[6] L. Militano, G. Araniti, M. Condoluci, I. Farris, and A. Iera. Device-to-Device Communications for 5G Internet of Things. EAI Endorsed Transactions on Internet of Things, 1(1): 150598, 2015. ISSN 2414-1399. doi: 10.4108/eai.26-10-2015.150598.

[7] I. S. H. Martínez, I. P. O. J. Salcedo, and I. B. S. R. Daza. IoT application of WSN on 5G infrastructure. 2017 International Symposium on Networks, Computers and Communications (ISNCC), pages 1–6, May 2017. doi: 10.1109/ISNCC.2017.8071989.

[8] Alex Vidigal Bastos and Diógenes Cecílio Da. Minicurso-Comunicação D2D para 5G de Arquiteturas de Redes Celulares: Da Teoria à Prática. https://www.researchgate.net. Accessed 2018-06-29.

[9] Udit Narayana Kar and Debarshi Kumar Sanyal. An overview of device-to-device communication in cellular networks. ICT express, 4(4):203–208, 2018.

[10] Furqan Jameel, Zara Hamid, Farhana Jabeen, Sherali Zeadally, and Muhammad Awais Javed. A survey of device-to-device communications: Research issues and challenges. IEEE Communications Surveys & Tutorials, 20(3):2133–2168, 2018.

[11] Francisco Rafael Marques Lima, Tarcisio Ferreira Maciel, and Francisco Rodrigo Porto Cavalcanti. Radio resource allocation in SC-FDMA uplink with resource adjacency constraints. Journal of Communication and Information Systems, 31(1), 2016.

[12] Kazi Ishfaq Ahmed, Hina Tabassum, and Ekram Hossain. Deep learning for radio resource allocation in multi-cell networks. IEEE Network, 33(6):188–195, 2019.

[13] G. Deng, J. Shi, G. Nie, and Z. Huang. Time-Varying Social-Aware Resource Allocation for Device-to-Device Communication. IEEE Access, 7:2653–2663, 2019. ISSN 2169-3536. doi: 10.1109/ACCESS.2018.2825244.

[14] Ghaith M. Jaradat, Anas Al-Badareen, Masri Ayob, Mutasem Al-Smadi, Ibrahim Al-Marashdeh, Mahmoud Ash-Shuqran, and Eyas Al-Odat. Hybrid Elitist-Ant System for Nurse-Rostering Problem. Journal of King Saud University - Computer and Information Sciences, 31(3):378 – 384, 2019. ISSN 1319-1578. doi: https://doi.org/10.1016/j.jksuci.2018.02.009.

[15] Wolfhard von Thienen and Tomer J. Czaczkes. Do Ants Use Ant Colony Optimization?, pages 265–291. Springer International Publishing, Cham, 2018. ISBN 978-3-319-77510-4. doi: 10.1007/978-3-319-77510-4_10.

[16] B. V. R. Gorantla and N. B. Mehta. Resource and Computationally Efficient Subchannel Allocation for D2D in Multi-Cell Scenarios With Partial and Asymmetric CSI. IEEE Transactions on Wireless Communications, 18(12):5806–5817, 2019.

[17] D. Feng, L. Lu, Y. Yuan-Wu, G. Y. Li, G. Feng, and S. Li. User selection based on limited feedback in device-to-device communications. In 2013 IEEE 24th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC), pages 2851–2855, 2013.

[18] D. Feng, L. Lu, Y. Yuan-Wu, G. Y. Li, G. Feng, and S. Li. Device-to-device communications underlaying cellular networks. IEEE Transactions on Communications, 61(8): 3541–3551, 2013.

[19] Yury Polyanskiy, H. Vincent Poor, and Sergio Verdu. Channel coding rate in the finite blocklength regime. IEEE Transactions on Information Theory, 56(5):2307–2359, 2010. doi: 10.1109/TIT.2010.2043769.

[20] Zheng Chu, Wenjuan Yu, Pei Xiao, Fuhui Zhou, Naofal Al-Dhahir, Atta ul Quddus, and Rahim Tafazolli. Opportunistic Spectrum Sharing for D2D-Based URLLC. IEEE Transactions on Vehicular Technology, 68(9):8995–9006, 2019. doi: 10.1109/TVT.2019.2931625.

[21] Rui Dong, Changyang She, Wibowo Hardjawana, Yonghui Li, and Branka Vucetic. Deep Learning for Radio Resource Allocation with Diverse Quality-of-Service Requirements in 5G. IEEE Transactions on Wireless Communications, 20(4):2309–2324, 2021. doi: 10.1109/TWC.2020.3041319.

[22] Haojun Yang, Kuan Zhang, Kan Zheng, and Yi Qian. Joint Frame Design and Resource Allocation for Ultra-Reliable and Low-Latency Vehicular Networks. IEEE Transactions on Wireless Communications, 19(5):3607–3622, 2020. doi: 10.1109/TWC.2020.2975576.

[23] Jing Cheng and Chao Shen. Relay selection and resource allocation for ultra-reliable uplink transmission in smart factory scenarios. arXiv preprint arXiv:2106.02677, 2021.

[24] Keshav Singh, Meng-Lin Ku, and Mark F. Flanagan. Resource Allocation in Energy-Efficient URLLC Multi-user Multicarrier AF Relay Networks. In ICC 2020 - 2020 IEEE International Conference on Communications (ICC), pages 1–6, 2020. doi: 10.1109/ICC40277.2020.9149182.

[25] Walid R. Ghanem, Vahid Jamali, Yan Sun, and Robert Schober. Resource Allocation for Multi-User Downlink URLLC-OFDMA Systems. In 2019 IEEE International Conference on Communications Workshops (ICC Workshops), pages 1–6, 2019. doi: 10.1109/ICCW.2019.8756746.

[26] 3GPP. 5G: Study on Scenarios and Requirements for Next Generation Access Technologies (Release 14). 3GPP TR 38.913 version 14.2.0 Release 14 document 38.913, 2017.

[27] Walid R Ghanem, Vahid Jamali, and Robert Schober. Optimal resource allocation for multi-user OFDMA-URLLC MEC systems. arXiv preprint arXiv:2009.11073, 2020.

[28] Idayat O. Sanusi, Karim M. Nasr, and Klaus Moessner. Radio Resource Management Approaches for Reliable Device-to-Device (D2D) Communication in Wireless Industrial Applications. IEEE Transactions on Cognitive Communications and Networking, 7(3):905–916, 2021. doi: 10.1109/TCCN.2020.3032679.

[29] I. O. Sanusi, K. M. Nasr, and K. Moessner. A Device to Device (D2D) Spectrum Sharing Scheme for Wireless Industrial Applications. In 2019 European Conference on Networks and Communications (EuCNC), pages 353–357, 2019. doi: 10.1109/EuCNC.2019.8802005.

[30] Hong Ren, Cunhua Pan, Yansha Deng, Maged Elkashlan, and Arumugam Nallanathan. Joint Power and Blocklength Optimization for URLLC in a Factory Automation Scenario. IEEE Transactions on Wireless Communications, 19(3):1786–1801, 2020. doi: 10.1109/TWC.2019.2957745.

[31] Helin Yang, Arokiaswami Alphones, Wen-De Zhong, Chen Chen, and Xianzhong Xie. Learning-Based Energy-Efficient Resource Management by Heterogeneous RF/VLC for Ultra-Reliable Low-Latency Industrial IoT Networks. IEEE Transactions on Industrial Informatics, 16(8):5565–5576, 2020. doi: 10.1109/TII.2019.2933867.

[32] Brandon Heller, Rob Sherwood, and Nick McKeown. The Controller Placement Problem. Proceedings of the First Workshop on Hot Topics in Software Defined Networks, pages 7–12, 2012. doi: 10.1145/2342441.2342444.

[33] Yannan Hu, Wang Wendong, Xiangyang Gong, X Que, and C Shiduan. Reliability-aware controller placement for Software-Defined Networks. In 2013 IFIP/IEEE International

Symposium on Integrated Network Management (IM 2013), pages 672–675, 2013. ISBN 1573-0077 VO -.

[34] Kshira Sagar Sahoo, Sampa Sahoo, Anamay Sarkar, Bibhudatta Sahoo, and Ratnakar Dash. On the placement of controllers for designing a wide area software defined networks. TENCON 2017 - 2017 IEEE Region 10 Conference, pages 3123–3128, 2017. doi: 10.1109/TENCON.2017.8228398.

[35] Alireza Farshin and Saeed Sharifian. MAP-SDN: a metaheuristic assignment and provisioning SDN framework for cloud datacenters. The Journal of Supercomputing, 73(9): 4112–4136, Sep 2017. ISSN 1573-0484. doi: 10.1007/s11227-017-2001-2.

[36] T. Wang, F. Liu, J. Guo, and H. Xu. Dynamic SDN controller assignment in data center networks: Stable matching with transfers. In IEEE INFOCOM 2016 - The 35th Annual IEEE International Conference on Computer Communications, pages 1–9, April 2016. doi: 10.1109/INFOCOM.2016.7524357.

[37] T. Wang, F. Liu, and H. Xu. An efficient online algorithm for dynamic sdn controller assignment in data center networks. IEEE/ACM Transactions on Networking, 25(5):2788–2801, Oct 2017. ISSN 1558-2566. doi: 10.1109/TNET.2017.2711641.

[38] M. Cello, Y. Xu, A. Walid, G. Wilfong, H. J. Chao, and M. Marchese. BalCon: A Distributed Elastic SDN Control via Efficient Switch Migration. In 2017 IEEE International Conference on Cloud Engineering (IC2E), pages 40–50, April 2017. doi: 10.1109/IC2E.2017.33.

[39] A. Filali, A. Kobbane, M. Elmachkour, and S. Cherkaoui. SDN Controller Assignment and Load Balancing with Minimum Quota of Processing Capacity. In 2018 IEEE International Conference on Communications (ICC), pages 1–6, May 2018. doi: 10.1109/ICC.2018.8422750.

[40] S. S. Savas, M. Tornatore, F. Dikbiyik, A. Yayimli, C. U. Martel, and B. Mukherjee. RASCAR: Recovery-Aware Switch-Controller Assignment and Routing in SDN. IEEE Transactions on Network and Service Management, 15(4):1222–1234, Dec 2018. ISSN 2373-7379. doi: 10.1109/TNSM.2018.2879865.

[41] D. Suh and S. Pack. Low-Complexity Master Controller Assignment in Distributed SDN Controller Environments. IEEE Communications Letters, 22(3):490–493, March 2018. ISSN 2373-7891. doi: 10.1109/LCOMM.2017.2787590.

[42] Bang Zhang, Xingwei Wang, and Min Huang. Dynamic controller assignment problem in software-defined networks. Transactions on Emerging Telecommunications Technologies, 29(8):e3460, 2018. doi: 10.1002/ett.3460.

[43] David Hock, Matthias Hartmann, Steffen Gebert, Michael Jarschel, Thomas Zinner, and Phuoc Tran-Gia. Pareto-optimal resilient controller placement in SDN-based core

networks. 25th International Teletraffic Congress (ITC), pages 1–9, sep 2013. doi: 10.1109/ITC.2013.6662939.

[44] Lingxia Liao and Victor CM Leung. Genetic algorithms with particle swarm optimization based mutation for distributed controller placement in sdns. In Network Function Virtualization and Software Defined Networks (NFV-SDN), 2017 IEEE Conference on, pages 1–6. IEEE, 2017.

[45] Vahid Ahmadi and Mostafa Khorramizadeh. An adaptive heuristic for multi-objective controller placement in software-defined networks. Computers and Electrical Engineering, 0: 1–25, 2017. ISSN 00457906. doi: 10.1016/j.compeleceng.2017.12.043.

[46] Md. Faizul Bari, Arup Raton Roy, Shihabur Rahman Chowdhury, Qi Zhang, Mohamed Faten Zhani, Reaz Ahmed, and Raouf Boutaba. Dynamic controller provisioning in software defined networks. In 2013 9th International Conference on Network and Service Management, CNSM 2013 and its three collocated Workshops - ICQT 2013, SVM 2013 and SETM 2013, pages 18–25. IEEE, oct 2013. ISBN 9783901882531. doi: 10.1109/CNSM.2013.6727805.

[47] T. Yuan, X. Huang, M. Ma, and J. Yuan. Balance-based SDN Controller Placement and Assignment with Minimum Weight Matching. In 2018 IEEE International Conference on Communications (ICC), pages 1–6, May 2018. doi: 10.1109/ICC.2018.8422637.

[48] H. K. Rath, V. Revoori, S. M. Nadaf, and A. Simha. Optimal controller placement in Software Defined Networks (SDN) using a non-zero-sum game. Proceeding of IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks 2014, WoWMoM 2014, pages 1–6, June 2014. doi: 10.1109/WoWMoM.2014.6918987.

[49] M. He, A. Basta, A. Blenk, and W. Kellerer. Modeling flow setup time for controller placement in SDN: Evaluation for dynamic flows. In 2017 IEEE International Conference on Communications (ICC), pages 1–7, May 2017. doi: 10.1109/ICC.2017.7996654.

[50] M. J. Abdel-Rahman, E. A. Mazied, K. Teague, A. B. MacKenzie, and S. F. Midkiff. Robust Controller Placement and Assignment in Software-Defined Cellular Networks. In 2017 26th International Conference on Computer Communication and Networks (ICCCN), pages 1–9, July 2017. doi: 10.1109/ICCCN.2017.8038439.

[51] Pavel Mach, Zdenek Becvar, and Tomas Vanek. In-Band Device-to-Device Communication in OFDMA Cellular Networks: A Survey and Challenges. IEEE Communications Surveys and Tutorials, 17(4):1885–1922, 2015. ISSN 1553877X. doi: 10.1109/COMST.2015.2447036.

[52] H. Sun, M. Sheng, X. Wang, Y. Zhang, J. Liu, and K. Wang. Resource allocation for maximizing the device-to-device communications underlaying LTE-Advanced networks. In 2013 IEEE/CIC International Conference on Communications in China - Workshops (CIC/ICCC), pages 60–64, Aug 2013. doi: 10.1109/ICCChinaW.2013.6670568.

[53] S. Ciou, J. Kao, C. Y. Lee, and K. Chen. Multi-sharing resource allocation for device-to-device communication underlaying 5G mobile networks. In 2015 IEEE 26th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC), pages 1509–1514, Aug 2015. doi: 10.1109/PIMRC.2015.7343537.

[54] H. H. Esmat, Mahmoud M. Elmesalawy, and I. I. Ibrahim. Adaptive Resource Sharing Algorithm for Device-to-Device Communications Underlaying Cellular Networks. IEEE Communications Letters, 20(3):530–533, 2016. ISSN 10897798. doi: 10.1109/LCOMM.2016.2517012.

[55] L. Wang, H. Tang, H. Wu, and G. L. Stüber. Resource Allocation for D2D Communications Underlay in Rayleigh Fading Channels. IEEE Transactions on Vehicular Technology, 66 (2):1159–1170, 2017.

[56] C. Kai, L. Xu, J. Zhang, and M. Peng. Joint Uplink and Downlink Resource Allocation for D2D Communication Underlying Cellular Networks. In 2018 10th International Conference on Wireless Communications and Signal Processing (WCSP), pages 1–6, 2018.

[57] Anum Ali, Ghalib A Shah, and Junaid Arshad. Energy efficient resource allocation for m2m devices in 5g. Sensors, 19(8):1830, 2019.

[58] Susan Dominic and Lillykutty Jacob. Distributed interference-aware admission control and resource allocation for underlaying D2D communications in cellular networks. Sādhanā, 44(6):138, 2019.

[59] Hongyuan Gao, Shibo Zhang, Yumeng Su, and Ming Diao. Joint resource allocation and power control algorithm for cooperative d2d heterogeneous networks. IEEE Access, 7: 20632–20643, 2019.

[60] Mengyuan Lee, Guanding Yu, and Geoffrey Ye Li. Accelerating resource allocation for d2d communications using imitation learning. In 2019 IEEE 90th Vehicular Technology Conference (VTC2019-Fall), pages 1–5. IEEE, 2019.

[61] Doyle Kwon and Duk Kim. Resource Allocation with a Rate Guarantee Constraint in Device-to-Device Underlaid Cellular Networks. Electronics, 9:438, 03 2020. doi: 10.3390/electronics9030438.

[62] Li Wang, Lu Liu, Xianghui Cao, Xiaohua Tian, and Yu Cheng. Sociality-aware resource allocation for device-to-device communications in cellular networks. IET Communications, 9(3):342–349, 2015.

[63] Zebing Feng, Zhiyong Feng, and T Aaron Gulliver. Effective small social community aware d2d resource allocation underlaying cellular networks. IEEE Wireless Communications Letters, 6(6):822–825, 2017.

[64] Gang Deng, Jiajiao Shi, Gaofeng Nie, and Zhaolong Huang. Time-varying social-aware resource allocation for device-to-device communication. IEEE Access, 7:2653–2663, 2018.

[65] De-Thu Huynh, Xiaofei Wang, Trung Q Duong, Nguyen-Son Vo, and Min Chen. Social-aware energy efficiency optimization for device-to-device communications in 5g networks. Computer Communications, 120:102–111, 2018.

[66] H. Yang, W. Zhong, C. Chen, A. Alphones, and X. Xie. Deep-Reinforcement-Learning-Based Energy-Efficient Resource Management for Social and Cognitive Internet of Things. IEEE Internet of Things Journal, 7(6):5677–5689, 2020.

[67] Nalin Jayaweera, Dileepa Marasinghe, Nandana Rajatheva, and Matti Latva-Aho. Factory Automation: Resource Allocation of an Elevated LiDAR System with URLLC Requirements. In 2020 2nd 6G Wireless Summit (6G SUMMIT), pages 1–5, 2020. doi: 10.1109/6GSUMMIT49458.2020.9083914.

[68] Ali Arshad Nasir, H. D. Tuan, Ha H. Nguyen, Mérouane Debbah, and H. Vincent Poor. Resource Allocation and Beamforming Design in the Short Blocklength Regime for URLLC. IEEE Transactions on Wireless Communications, 20(2):1321–1335, 2021. doi: 10.1109/TWC.2020.3032729.

[69] Bikramjit Singh, Zexian Li, and Mikko A. Uusitalo. Flexible resource allocation for device-to-device communication in FDD system for ultra-reliable and low latency communications. In 2017 Advances in Wireless and Optical Communications (RTUWO), pages 186–191, 2017. doi: 10.1109/RTUWO.2017.8228531.

[70] Federico Librino and Paolo Santi. Resource Allocation and Sharing in URLLC for IoT Applications Using Shareability Graphs. IEEE Internet of Things Journal, 7(10):10511–10526, 2020. doi: 10.1109/JIOT.2020.2999645.

[71] Federico Librino and Paolo Santi. The Complexity–Performance Tradeoff in Resource Allocation for URLLC Exploiting Dynamic CSI. IEEE Internet of Things Journal, 8(17): 13266–13277, 2021. doi: 10.1109/JIOT.2021.3066104.

[72] Helin Yang, Xianzhong Xie, and Michel Kadoch. Intelligent Resource Management Based on Reinforcement Learning for Ultra-Reliable and Low-Latency IoV Communication Networks. IEEE Transactions on Vehicular Technology, 68(5):4157–4169, 2019. doi: 10.1109/TVT.2018.2890686.

[73] Yan Wu, Dan Wu, Liang Ao, Lianxin Yang, and Qinxue Fu. Contention-Based Radio Resource Management for URLLC-Oriented D2D Communications. IEEE Transactions on Vehicular Technology, 69(9):9960–9971, 2020. doi: 10.1109/TVT.2020.3003944.

[74] T. Schwengler. Wireless and Cellular Communications (paperback). Lulu.com, 2019. ISBN 9780359570591. URL `https://books.google.com.br/books?id=afdbzQEACAAJ`.

[75] H. Sun, M. Sheng, X. Wang, Y. Zhang, J. Liu, and K. Wang. Resource allocation for maximizing the device-to-device communications underlaying LTE-Advanced networks.

In 2013 IEEE/CIC International Conference on Communications in China - Workshops (CIC/ICCC), pages 60–64, Aug 2013. doi: 10.1109/ICCChinaW.2013.6670568.

[76] S. Ciou, J. Kao, C. Y. Lee, and K. Chen. Multi-sharing resource allocation for device-to-device communication underlaying 5G mobile networks. In 2015 IEEE 26th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC), pages 1509–1514, Aug 2015. doi: 10.1109/PIMRC.2015.7343537.

[77] Kuo-Yi Chen, Jung-Chun Kao, Si-An Ciou, and Shih-Han Lin. Joint spectrum reuse and power control for multi-sharing device-to-device communication. arXiv preprint arXiv:1702.06780, 2017.

[78] Xu-ping Zhai, Gong Lei, and Nan Zhang. Resource Allocation and Power Control for D2D Communication Underlaying LTE-Advanced Networks. Transactions on Computer Science and Engineering, (cnai), 2018.

[79] Yao-Jen Liang and Yi-Shao Lin. A non-iterative resource allocation strategy for device-to-device communications in underlaying cellular networks. Wireless Networks, 23(8): 2485–2497, Nov 2017. ISSN 1572-8196. doi: 10.1007/s11276-016-1302-3.

[80] Mengmeng Ru, Sixing Yin, and Zhaowei Qu. Power and Spectrum Allocation in D2D Networks Based on Coloring and Chaos Genetic Algorithm. Procedia Computer Science, 107: 183 – 189, 2017. ISSN 1877-0509. doi: https://doi.org/10.1016/j.procs.2017.03.076. Advances in Information and Communication Technology: Proceedings of 7th International Congress of Information and Communication Technology (ICICT2017).

[81] S. Cicalò and V. Tralli. QoS-Aware Admission Control and Resource Allocation for D2D Communications Underlaying Cellular Networks. IEEE Transactions on Wireless Communications, 17(8):5256–5269, Aug 2018. ISSN 1536-1276. doi: 10.1109/TWC. 2018.2840141.

[82] C. Li, F. Jiang, X. Wang, and B. Shen. Optimal relay selection based on social threshold for D2D communications underlay cellular networks. pages 1–6. 2016 8th International Conference on Wireless Communications Signal Processing (WCSP), Oct 2016. doi: 10. 1109/WCSP.2016.7752453.

[83] X. Li, X. Liu, Y. Li, L. Xiao, and J. Wang. A cooperative rate splitting scheme for device-to-device underlay cellular communication. In 2017 23rd Asia-Pacific Conference on Communications (APCC), pages 1–5, 2017.

[84] Sara Sharifi and Mohammad Fathi. Underlay device to device communication with imperfect interference channel knowledge. Wireless Personal Communications, 101(2):619–634, 2018.

[85] Y. P. Llerena and P. R. L. Gondim. SDN-Controller Placement for D2D Communications. IEEE Access, 7:169745–169761, 2019. ISSN 2169-3536. doi: 10.1109/ACCESS.2019. 2955434.

[86] Abdulkadir Celik, Redha M. Radaydeh, Fawaz S. Al-Qahtani, and Mohamed-Slim Alouini. Resource Allocation and Interference Management for D2D-Enabled DL/UL Decoupled Het-Nets. IEEE Access, 5:22735–22749, 2017. doi: 10.1109/ACCESS.2017.2760350.

[87] Asmaa Abdallah, Mohammad M. Mansour, and Ali Chehab. Power Control and Channel Allocation for D2D Underlaid Cellular Networks. IEEE Transactions on Communications, 66(7):3217–3234, 2018. doi: 10.1109/TCOMM.2018.2812731.

[88] UmaMaheswari Gurusamy, Hariharan K., and Manikandan M.S.K. Modelling and Performance Analysis of Flow Management in a Multi-Controller Software Defined Network using M/M/c/K model. In 2018 15th IEEE India Council International Conference (INDICON), pages 1–5, 2018. doi: 10.1109/INDICON45594.2018.8987106.

[89] Strahil Panev and Pero Latkoski. The Impact of the Most Common Assumptions When Modelling SDN-based Mobile Networks. Microwave Review, 25(2), 2019.

[90] Bo Li, Xiaoheng Deng, and Yiqin Deng. Mobile-edge computing-based delay minimization controller placement in SDN-IoV. Computer Networks, 193:108049, 2021. ISSN 1389-1286. doi: https://doi.org/10.1016/j.comnet.2021.108049.

[91] John F. Shortle, James M. Thompson, Donald Gross, and Carl M. Harris. Fundamentals of Queueing Theory. John Wiley & Sons, New Jersey, USA, 5th edition, 2018. ISBN 978-0-471-79127-0. doi: 10.1017/CBO9781107415324.004.

[92] S Almufti, R Marqas, and V Ashqi. Taxonomy of bio-inspired optimization algorithms. Journal Of Advanced Computer Science & Technology, 8(2):23, 2019.

[93] Jing Wang and Gerardo Beni. Cellular robotic system with stationary robots and its application to manufacturing lattices. In Proceedings. IEEE International Symposium on Intelligent Control 1989, pages 132–137. IEEE, 1989.

[94] Jun Tang, Gang Liu, and Qingtao Pan. A Review on Representative Swarm Intelligence Algorithms for Solving Optimization Problems: Applications and Trends. IEEE/CAA Journal of Automatica Sinica, 8(10):1627–1643, 2021. doi: 10.1109/JAS.2021.1004129.

[95] A Hanif Halim and IJAoCMiE Ismail. Combinatorial optimization: comparison of heuristic algorithms in travelling salesman problem. Archives of Computational Methods in Engineering, 26(2):367–380, 2019.

[96] David H Wolpert and William G Macready. No free lunch theorems for optimization. IEEE transactions on evolutionary computation, 1(1):67–82, 1997.

[97] Xin-She Yang. Swarm-based metaheuristic algorithms and no-free-lunch theorems. Theory and new applications of swarm intelligence, 9:1–16, 2012.

[98] Dervis Karaboga, Beyza Gorkemli, Celal Ozturk, and Nurhan Karaboga. A comprehensive survey: artificial bee colony (ABC) algorithm and applications. Artificial Intelligence Review, 42(1):21–57, Jun 2014. ISSN 1573-7462. doi: 10.1007/s10462-012-9328-0.

[99] Dervis Karaboga and Beyza Gorkemli. Solving Traveling Salesman Problem by Using Combinatorial Artificial Bee Colony Algorithms. International Journal on Artificial Intelligence Tools, 28(01):1950004, 2019.

[100] Quan-Ke Pan, M. Fatih Tasgetiren, P.N. Suganthan, and T.J. Chua. A discrete artificial bee colony algorithm for the lot-streaming flow shop scheduling problem. Information Sciences, 181(12):2455 – 2468, 2011. ISSN 0020-0255. doi: https://doi.org/10.1016/j.ins.2009.12.025.

[101] Marco Dorigo and Thomas Stützle. Ant Colony Optimization: Overview and Recent Advances, pages 311–351. Springer International Publishing, Cham, 2019. ISBN 978-3-319-91086-4. doi: 10.1007/978-3-319-91086-4_10.

[102] Franco Luis Alejandro Arito and San Luis-Argentina. Algoritmos de Optimización basados en Colonias de Hormigas aplicados al Problema de Asignación Cuadrática y otros problemas relacionados. Universidad Nacional de San Luis Facultad de Ciencias Físico Matemáticas y Naturales Departamento de Informática, San Luis–Argentina Abril de, 2010.

[103] Russell Eberhart and James Kennedy. A new optimizer using particle swarm theory. In MHS'95. Proceedings of the sixth international symposium on micro machine and human science, pages 39–43. IEEE, 1995.

[104] Wei-Neng Chen, Jun Zhang, Henry SH Chung, Wen-Liang Zhong, Wei-Gang Wu, and Yuhui Shi. A novel set-based particle swarm optimization method for discrete optimization problems. IEEE Transactions on evolutionary computation, 14(2):278–300, 2009.

[105] Long Yao, Peilin Hong, Wen Zhang, Jianfei Li, and Dan Ni. Controller placement and flow based dynamic management problem towards SDN. 2015 IEEE International Conference on Communication Workshop, ICCW 2015, 2015:363–368, 2015. doi: 10.1109/ICCW.2015.7247206.

[106] Stanislav Lange, Steffen Gebert, Thomas Zinner, Phuoc Tran-Gia, David Hock, Michael Jarschel, and Marco Hoffmann. Heuristic approaches to the controller placement problem in large scale sdn networks. IEEE Transactions on Network and Service Management, 12(1):4–17, 2015.

[107] Rajendra Jain, Dah-Ming Chiu, and William R. Hawe. A quantitative measure of fairness and discrimination for resource allocation in shared computer system. DEC technical report TR301, cs.NI/9809(DEC-TR-301):1–38, 1984. ISSN 07338716. doi: 10.1109/JSAC.2003.809458.

[108] Alireza Farshin and Saeed Sharifian. A chaotic grey wolf controller allocator for Software Defined Mobile Network (SDMN) for 5th generation of cloud-based cellular systems (5G). Computer Communications, 108:94 – 109, 2017. ISSN 0140-3664. doi: 10.1016/j.comcom.2017.05.003.

[109] Kamal Jain, Jitendra Padhye, Venkata N. Padmanabhan, and Lili Qiu. Impact of Interference on Multi-Hop Wireless Network Performance. Wireless Networks, 11(4):471–487, Jul 2005. ISSN 1572-8196. doi: 10.1007/s11276-005-1769-9.

[110] X. Yi and G. Caire. Topological Interference Management With Decoded Message Passing. IEEE Transactions on Information Theory, 64(5):3842–3864, May 2018. doi: 10.1109/TIT.2018.2792045.

[111] C. Xu, C. Gao, Z. Zhou, Z. Chang, and Y. Jia. Social network-based content delivery in device-to-device underlay cellular networks using matching theory. IEEE Access, 5: 924–937, 2017. ISSN 2169-3536. doi: 10.1109/ACCESS.2016.2621010.

[112] Wei-neng Chen, Student Member, Jun Zhang, Senior Member, Henry S H Chung, and Senior Member. A Novel Set-Based Particle Swarm Optimization Method for Discrete Optimization Problems. IEEE Transactions on Evolutionary Computation, 14(2):278–300, 2010. ISSN 1089-778X. doi: 10.1109/TEVC.2009.2030331.

[113] Kandethody M Ramachandran and Chris P Tsokos. Mathematical Statistics with Applications. Elsevier, 2009.

[114] Dennis Wackerly, William Mendenhall, and Richard L Scheaffer. Mathematical statistics with applications. Cengage Learning, 2014.

[115] Julian McAuley and Jure Leskovec. Learning to Discover Social Circles in Ego Networks. NIPS, 2012, https://snap.stanford.edu/data/ego-Facebook.html. Accessed 2019-04-01.

A - COMPUTATIONAL COMPLEXITY OF THE METAHEURISTICS (CPP PROBLEM).

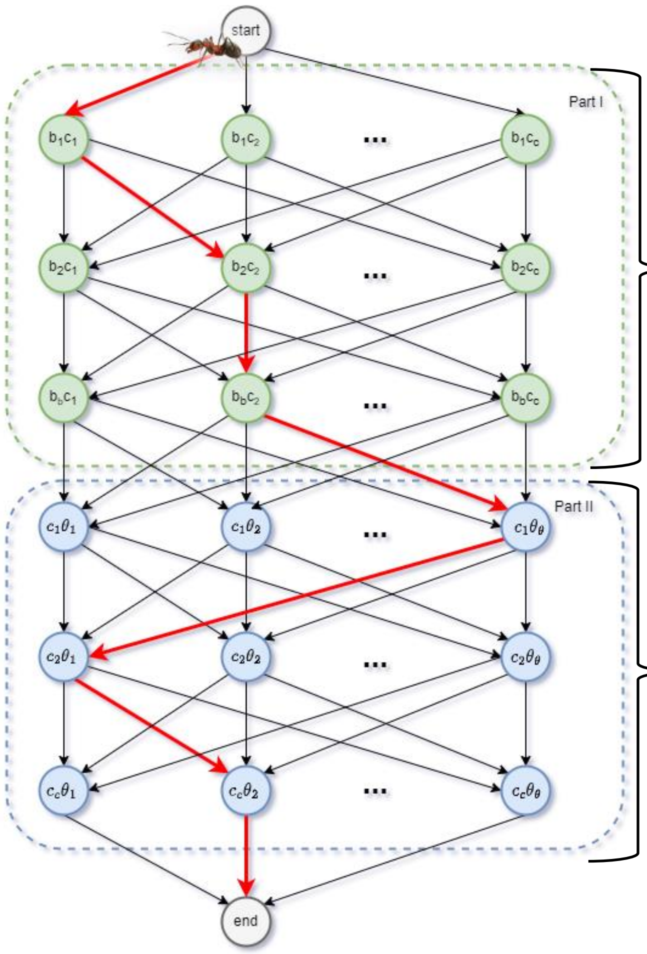
This appendix presents a detailed discussion about the computational complexity of the ABC, ACS-EM, ACS and PSO algorithms applied to CPP problem. We have defined the following sets for a better understanding of such complexity:

$S = \{s_1, s_2, \ldots, s_s\}$ of SGW (Serving Gateway)

$B = \{b_1, b_2, \ldots, b_b\}$ of eNBs (evolved NodeB)

$C = \{c_1, c_2, \ldots, c_c\}$ of controllers that can be in any entity in the cellular network (PGW, SGW or eNB)

$\Theta = PGW \cup \underbrace{\{s_1, s_2, \ldots, s_s\}}_{SGW} \cup \underbrace{\{b_1, b_2, \ldots, b_b\}}_{eNB} = \{\theta_1, \theta_2, \ldots, \theta_\theta\}$ was defined as the possible locations of the controllers.



$$V_{b_k c_j} = \begin{bmatrix} b_1c_1 & \cdots & b_1c_c \\ \vdots & \vdots & \vdots \\ b_bc_1 & \cdots & b_bc_c \end{bmatrix} \Rightarrow bc$$

$$V_{c_j \theta_i} = \begin{bmatrix} c_1\theta_1 & \cdots & c_1\theta_\theta \\ \vdots & \vdots & \vdots \\ c_c\theta_1 & \cdots & c_c\theta_\theta \end{bmatrix} \Rightarrow c\theta$$

Below is the discussion about the computational complexity of the four algorithms.

First, we calculated the computational complexity of the main functions used in the Ant Colony System with External Memory (ACS-EM) and Ant Colony System (ACS) algorithms.

InitPheromone()

```
Input: τ0
for each node ∈ node1 do → O(bc)
    for each child ∈ node.Children do → O(c)        } → O(bc²)
        τ[node, child] = τ0
    end for
end for
for each node ∈ node2 do → O(cθ)
    for each child ∈ node.Children do → O(θ)        } → O(cθ²)
        τ[node, child] = τ0
    end for
end for
Output: τ[·]
```

$\rightarrow O(bc^2) + O(c\theta^2) = O(c\theta^2)$ *since* $\theta > b \geq c$

InitMemory()

for each $node \in node_1$ **do** → $\boldsymbol{O(bc)}$
  $memory[node] = 0$
end for
for each $node \in node_2$ **do** → $\boldsymbol{O(c\theta)}$
  $memory[node] = 0$
end for
Output: $memory[\cdot]$

→ $\boldsymbol{O(bc) + O(c\theta) = O(c\theta)}$ ***since*** $\boldsymbol{\theta > b \geq c}$

CalculateChildrenHeuristic()

if $currentPosition \in node_1$ **then**
  for each $child \in currentPosition.Children$ **do** → $\boldsymbol{O(c)}$
    $\eta[child] = GetHeuristicValue(child)$
  end for
else
  for each $child \in currentPosition.Children$ **do** → $\boldsymbol{O(\theta)}$
    $\eta[child] = GetHeuristicValue(child)$
  end for
end if
Output: $\eta[\cdot]$

→ $\boldsymbol{O(c) + O(\theta) = O(\theta)}$ ***since*** $\boldsymbol{\theta > c}$

CalculateChildrenProbability()

Input: α, β, τ, η
if $currentPosition \in node_1$ **then**
  for each $child \in currentPosition.Children$ **do** → $\boldsymbol{O(c)}$
    $p_k[child] = GetProbability(child)$
  end for
else
  for each $child \in currentPosition.Children$ **do** → $\boldsymbol{O(\theta)}$
    $p_k[child] = GetProbability(child)$
  end for
end if
Output: $p_k[\cdot]$

→ $\boldsymbol{O(c) + O(\theta) = O(\theta)}$ ***since*** $\boldsymbol{\theta > c}$

OfflinePheromoneUpdate()

Input: δ, BestTour
for each $node \in node_1$ **do** → $\boldsymbol{O(bc)}$
  for each $child \in node.Children$ **do** → $\boldsymbol{O(c)}$
    $\tau[node, child] = GetOfflinePheromoneValue()$
  end for
end for
for each $node \in node_2$ **do** → $\boldsymbol{O(c\theta)}$
  for each $child \in node.Children$ **do** → $\boldsymbol{O(\theta)}$
    $\tau[node, child] = GetOfflinePheromoneValue()$
  end for
end for
Output: $\tau[\cdot]$

→ $\boldsymbol{O(bc^2)}$

→ $\boldsymbol{O(c\theta^2)}$

→ $\boldsymbol{O(bc^2) + O(c\theta^2) = O(c\theta^2)}$ ***since*** $\boldsymbol{\theta > b \geq c}$

UpdateMemory()

Input: BestTour
for each $node \in BestTour$ **do**
  $memory[node] += 1$
end for
Output: $memory[\cdot]$

→ $\boldsymbol{O(b + c) = O(b)}$ ***since*** $\boldsymbol{b \geq c}$

ACS-EM - Ant Colony System with External Memory

```
Input: Network_Topology, C_D2D, Queuing_Parameters, τ0, α, β, q0, r0, z0, δ, ant_quantity, iteration_max
  ▷ inicialization
τ[·] = InitPheromone() → O(cθ²)                                    } → O(cθ²) + O(cθ) = O(cθ²)
memory[·] = InitMemory() → O(cθ)                                   }
for i = 0 to iteration_max do → O(iteration_max)
  for each a ∈ ant_quantity do → O(ant_quantity)
    currentPosition ← RootNode → O(1)
    TourList ← currentPosition → O(1)
    while currentPosition.Children ≠ 0 do → O(b + c) = O(b) since b ≥ c
      η[·] ← CalculateChildrenHeuristic() → O(θ)
      q ← RandomDouble() → O(1)
      if q < q0 then
        r ← RandomDouble() → O(1)
        if r < r0 then
          nextPosition ← arg_máx{memory[child]} → O(c) ∨ O(θ) → O(θ)
        else
          nextPosition ← arg_máx{τ[child] * η[child]^β} → O(c) ∨ O(θ) → O(θ)
        end if
      else                                                            → O(θ)      → O(b) * [O(θ) + O(θ) + O(b)]
        z ← RandomDouble() → O(1)                                                    = O(b) * O(θ)
        if z < z0 then                                                               = O(bθ)
          nextPosition ← arg_mín{memory[child]} → O(c) ∨ O(θ) → O(θ)
        else
          p_z[·] ← CalculateChildrenProbability() → O(θ)
          nextPosition ← Get_Z(p_z) → O(c) ∨ O(θ) → O(θ)
        end if
      end if
      Add nextPosition to TourList → O(1)
      ValidateFeasiblePath(TourList) → O(b + c) = O(b) since b ≥ c
      currentPosition ← nextPosition → O(1)
      LocalPheromoneUpdate() → O(1)
    end while
    if CalculateAntsTour() < BestTour then
      BestTour ← TourList → O(1)
    end if
  end for
  OfflinePheromoneUpdate(BestTour) → O(cθ²)                       } → O(cθ²) + O(b) = O(cθ²)
  memory[·] = UpdateMemory(BestTour) → O(b)                        }
  if BestTour < GlobalBestTour then
    GlobalBestTour ← BestTour → O(1)
  end if
end for
Output: GlobalBestTour
```

Therefore, the computational complexity of ACS-EM algorithm is:

$$\boldsymbol{O(c\theta^2) + O(iteration\_max) * [O(ant\_quantity) * O(b\theta) + O(c\theta^2)]}$$

$$\boldsymbol{= O(iteration\_max) * [O(ant\_quantity) * O(b\theta) + O(c\theta^2)]}$$

$$\boldsymbol{= O(iteration\_max * [ant\_quantity * b\theta + c\theta^2])}$$

ACS - Ant Colony System

Input: $Network\_Topology, C_{D2D}, Queuing\_Parameters, \tau_0, \beta, q_0, \delta, ant\_quantity, iteration\_max$

▷ inicialization

$\tau[\cdot] = InitPheromone() \rightarrow \boldsymbol{O(c\theta^2)}$
for $i = 0$ to iteration_max **do** $\rightarrow \boldsymbol{O(iteration\_max)}$
  for each $m \in ant\_quantity$ **do** $\rightarrow \boldsymbol{O(ant\_quantity)}$
    $currentPosition \leftarrow RootNode \rightarrow \boldsymbol{O(1)}$
    $TourList \leftarrow currentPosition \rightarrow \boldsymbol{O(1)}$
    while $currentPosition.Children \neq 0$ **do** $\rightarrow \boldsymbol{O(b+c) = O(b)\ since\ b \geq c}$
      $\eta[\cdot]$ = CalculateChildrenHeuristic() $\rightarrow \boldsymbol{O(\theta)}$
      $q \leftarrow RandomDouble() \rightarrow \boldsymbol{O(1)}$
      if $q \leq q_o$ **then**
        $nextPosition \leftarrow \arg\max_{j \in C^k}\{\tau(i,j)^\alpha \cdot \eta(i,j)^\beta\} \rightarrow \boldsymbol{O(c)} \vee \boldsymbol{O(\theta)} \rightarrow \boldsymbol{O(\theta)}$
      else
        $p_k[\cdot] \leftarrow CalculateChildrenProbability() \rightarrow \boldsymbol{O(\theta)}$
        $r \leftarrow RandomDouble() \rightarrow \boldsymbol{O(1)}$
        $nextPosition \leftarrow GetChildrenByRouletteMethod(p, r) \rightarrow \boldsymbol{O(c)} \vee \boldsymbol{O(\theta)} \rightarrow \boldsymbol{O(\theta)}$
      end if
      Add nextPosition to TourList $\rightarrow \boldsymbol{O(1)}$
      $ValidateFeasiblePath(TourList) \rightarrow \boldsymbol{O(b+c) = O(b)\ since\ b \geq c}$
      $currentPosition \leftarrow nextPosition \rightarrow \boldsymbol{O(1)}$
      LocalPheromoneUpdate() $\rightarrow \boldsymbol{O(1)}$
    end while
    if $CalculateAntsTour() < BestTour$ **then**
      $BestTour \leftarrow TourList \rightarrow \boldsymbol{O(1)}$
    end if
  end for
  $OfflinePheromoneUpdate(BestTour) \rightarrow \boldsymbol{O(c\theta^2)}$
  if $BestTour < GlobalBestTour$ **then**
    $GlobalBestTour \leftarrow BestTour \rightarrow \boldsymbol{O(1)}$
  end if
end for

Output: GlobalBestTour

$\rightarrow \boldsymbol{O(\theta)}$

$\rightarrow \boldsymbol{O(b)} * [\boldsymbol{O(\theta)} + \boldsymbol{O(\theta)} + \boldsymbol{O(b)}] = \boldsymbol{O(b)} * \boldsymbol{O(\theta)} = \boldsymbol{O(b\theta)}$

The computational complexity of ACS is

$$\boldsymbol{O(c\theta^2) + O(iteration\_max) * [O(ant\_quantity) * O(b\theta) + O(c\theta^2)]}$$
$$\boldsymbol{= O(iteration\_max) * [O(ant\_quantity) * O(b\theta) + O(c\theta^2)]}$$
$$\boldsymbol{= O(iteration\_max * [ant\_quantity * b\theta + c\theta^2])}$$

Finally, is the calculation of the computational complexity of the main functions used in PSO.

InitVelocity()

for each $node \in node_1$ **do** $\rightarrow \boldsymbol{O(bc)}$
  $node.SetInitVelocity()$
end for
for each $node \in node_2$ **do** $\rightarrow \boldsymbol{O(c\theta)}$
  $node.SetInitVelocity()$
end for

Output: $velocity[\cdot]$

$\rightarrow \boldsymbol{O(bc) + O(c\theta) = O(c\theta)}$

InitPosition()

Input: particle_quantity
for each $m \in particle\_quantity$ **do** $\rightarrow \boldsymbol{O(particle\_quantity)}$
  for each $b \in node_1$ **do** $\rightarrow \boldsymbol{O(b)}$
    $position[m].Add(Select\_Random(node_1[b])) \rightarrow \boldsymbol{O(1)}$
  end for
  for each $c \in node_2$ **do** $\rightarrow \boldsymbol{O(c)}$
    $position[m].Add(Select\_Random(node_2[c])) \rightarrow \boldsymbol{O(1)}$
  end for
end for

Output: $position[\cdot]$

$\rightarrow \boldsymbol{O(b) + O(c) = O(b)}$

$\rightarrow \boldsymbol{O(particle\_quantity * b)}$

velocity_updating()

```
Input: w_max; w_min; c1; c2; BestSoFar; GlobalBestPosition
▷ V ← w * V + c1 * r1 * (BestSoFar − X) + c2 * r2 * (GlobalBestPosition − X)
r1 ← RandomValue() → O(1)
r2 ← RandomValue() → O(1)
pbest ← Position.Substract(BestSoFar, currentPosition) → O(b + c) = O(b)
gbest ← Position.Substract(GlobalBestPosition, currentPosition) → O(b + c) = O(b)
for each b ∈ node_1 do → O(b)
    v_1 ← Multiply(c1 * r1, pbest[b]) → O(1)
    v_2 ← Multiply(c2 * r2, gbest[b]) → O(1)
    v_3 ← Multiply(w, currentPosition[b]) → O(1)        → O(b)
    V_b ← Sum(v_1, v_2, v_3) → O(1)
end for                                                   → O(b)
for each c ∈ node_2 do → O(c)
    v_1 ← Multiply(c1 * r1, pbest[c]) → O(1)
    v_2 ← Multiply(c2 * r2, gbest[c]) → O(1)
    v_3 ← Multiply(w, currentPosition[c]) → O(1)        → O(c)
    V_c ← Sum(v_1, v_2, v_3) → O(1)
end for
```

position_updating()

```
Input: w_max; w_min; c1; c2, particle_quantity, iteration_max
r ← RandomValue()
for each b ∈ node_1 do → O(b)
    for each node ∈ Candidate_Set_b do → O(c)
        if velocity(node) ≥ r then
            X_m.Add(node) → O(1)                          → O(bc)
            break
        end if
    end for
end for                                                   → O(cθ) since θ > b ≥ c
for each c ∈ node_2 do → O(c)
    for each node ∈ Candidate_Set_c do → O(θ)
        if velocity(node) ≥ r then
            X_m.Add(node) → O(1)                          → O(cθ)
            break
        end if
    end for
end for
Output: X_m
```

PSO - Particle Swarm Optimization Algorithm

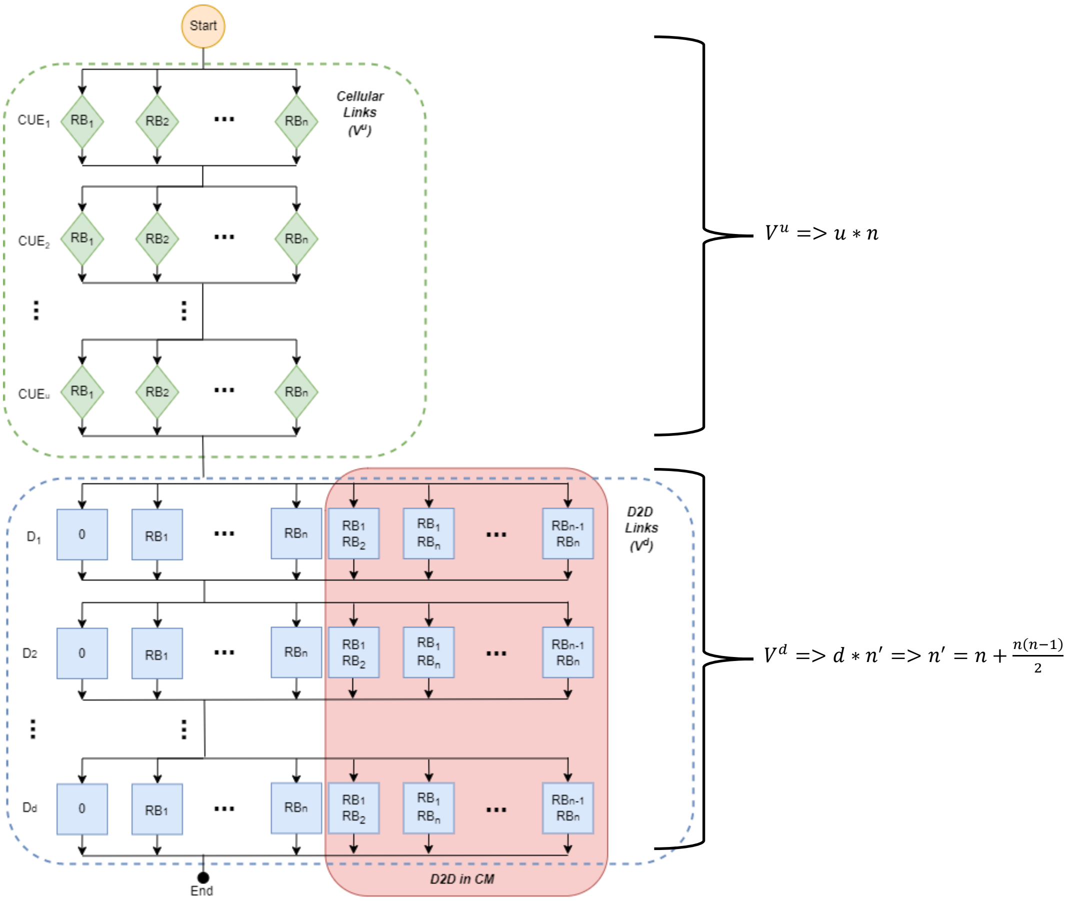
```
Input: w_max; w_min; c1; c2, particle_quantity, iteration_max
▷ inicialization
InitVelocity() → O(cθ)
InitPosition() → O(particle_quantity * b)                 → O(cθ) + O(particle_quantity * b)
for i = 0 to iteration_max do → O(iteration_max)
    for each m ∈ particle_quantity do → O(particle_quantity)
        velocity_updating() → O(b)
        X_m = position_updating() → O(cθ)
        ValidateFeasiblePatch(X_m) → O(b + c) = O(b)
        currentPositition ← X_m → O(1)                    → O(cθ)
        if X_m < BestSoFar then                           → O(iteration_max) * O(particle_quantity) * O(cθ)
            BestSoFar ← X_m → O(1)
        end if
    end for
    if BestSoFar < GlobalBestPosition then
        GlobalBestPosition ← BestSoFar → O(1)
    end if
end for
Output: GlobalBestPosition
```

Therefore, the computational complexity of PSO is:

$$\boldsymbol{O(c\theta) + O(particle\_quantity * b) + O(iteration\_max) * O(particle\_quantity) * O(c\theta)}$$
$$\boldsymbol{= O(iteration\_max) * O(particle\_quantity) * O(c\theta)}$$

B - COMPUTATIONAL COMPLEXITY OF THE METAHEURISTICS (RRA PROBLEM).

This appendix presents a detailed discussion about computational complexity of the ABC, ACS-EM, and PSO algorithms applied to RRA problem.



In what follows is a discussion on the computational complexity of the three optimization solutions.

The computational complexity of ABC is $O(nro\_it * SN * (SN + u + d + n))$

Algorithm 1. *Artificial Bee Colony (ABC)*

```
Input: SN, limit, NL_size, u, d, N_u, nro_it, α1, α2, α3, α4
n ← u + N_u
CLinks ← Create_CLinks(u) → O(u)
D2DLinks ← Create_D2DLinks(d) → O(d)
AllSolutions[SN] ← InitPopulation(SN, N, CLinks, D2DLinks, α1, α2, α3, α4) → O(SN * (u + d + n))
for i = 0 to nro_it do                                              } O(nro_it * SN * (SN + u + d + n))

***Employed bee phase***
  for s = 0 to SN do                                                } O(SN * (u + d + n))
    if OperatorNeighborList == 0 then
      GenerateOperatorNeighborList(NL_size)
    end if
    solution ← AllSolutions[s]
    newSolution ← GetFeasibleNeighbor(solution, OperatorNeighborList.First, N) → O(n)
    Fitnes_newSolution ← CalculateOF(newSolution, α1, α2, α3, α4) → O(u + d)
    if Fitnes_newSolution ≥ Fitness[solution] then
      AllSolutions[s] ← newSolution
      Fitness[s] ← Fitnes_newSolution
      WNL ← OperatorNeighborList.First
      Limit[solution] ← 0
    else
      Limit[solution] ← Limit[solution] + 1
    end if
    RemoveOperatorNeighborList.First
  end for

***Onlooker bee phase***
  for s = 0 to SN do                                                } O(SN * (SN + u + d + n))
    solution ← ProbabilisticRouletteOperator(Fitness[SN]) → O(SN)
    if OperatorNeighborList == 0 then
      GenerateOperatorNeighborList(NL_size)
    end if
    newSolution ← GetFeasibleNeighbor(solution, OperatorNeighborList.First, N) → O(n)
    Fitnes_newSolution ← CalculateOF(newSolution, α1, α2, α3, α4) → O(u + d)
    if Fitnes_newSolution ≥ Fitness[solution] then
      Solutions[solution] ← newSolution
      Fitness[solution] ← Fitnes_newSolution
      WNL ← OperatorNeighborList.First
      Limit[solution] ← 0
    else
      Limit[solution] ← Limit[solution] + 1
    end if
    RemoveOperatorNeighborList.First

    LocalBestSolution ← GetBestSolution(AllSolutions[SN]) → O(SN)
    LocalBestFitness ← GetBestFitness(Fitness[SN]) → O(SN)
    if LocalBestFitness > GlobalBestFitness then
      GlobalBestSolution ← LocalBestSolution
      GlobalBestFitness ← LocalBestFitness
    end if
  end for

***Scout bee phase***
  higherLimitSolution ← GetHigherLimitSolution() → O(SN)
  AllSolutions[higherLimitSolution] ← GetRandomFeasibleSolution(N, CLinks, D2DLinks) → O(u + d + n)
  solution ← AllSolutions[higherLimitSolution]
  Fitness[solution] ← CalculateOF(solution, α1, α2, α3, α4) → O(u + d)
  Limit[solution] ← 0
  for s = 0 to SN do                                                } O(SN * (u + d + n))
    if Limit[s] ≥ limit then
      solution ← GetRandomFeasibleSolution(N, CLinks, D2DLinks) → O(u + d + n)
      AllSolutions[s] ← solution
      Fitness[s] ← CalculateOF(solution, α1, α2, α3, α4) → O(u + d)
      Limit[s] ← 0
    end if
  end for
end for
Output: GlobalBestSolution (allocation, transmit power and communication mode)
```

InitPopulation()

```
Input: SN, N, CLinks, D2DLinks, α1, α2, α3, α4
for s = 0 to SN do                                                  } O(SN * (u + d + n))
  foodSource = GetRandomFeasibleSolution(N, CLinks, D2DLinks) → O(u + d + n)
  AllSolutions[s] ← foodSource
  Fitness[s] ← CalculateOF(foodSource, α1, α2, α3, α4) → O(u + d)
  Limit[s] ← 0
end for
Output: AllSolutions[·]
```

GetFeasibleNeighbor()

```
Input: solution, operator, N                                        } O(n)
neighbor ← ApplyOperator(solution, operator) → O(1)
ValidateFeasibleSolution(N, neighbor) → O(n)
Output: neighbor
```

GetRandomFeasibleSolution()

```
Input: N, CLinks, D2DLinks                                          } O(u + d + n)
for each clink ∈ CLinks do                                          } O(u)
  foodSource.CLink_Allocation[clink] = Select_Random_RB(N)
  foodSource.CLink_TransmissionPower[clink] = Select_Random_Power()
end for
for each dlink ∈ D2DLinks do                                        } O(d)
  foodSource.DLink_Allocation[dlink] = Select_Random_RB(N)
  foodSource.DLink_TransmissionPower[dlink] = Select_Random_Power()
  foodSource.DLink_CommunicationMode[dlink] = Select_CommunicationMode()
end for
ValidateFeasibleSolution(N, foodSource) → O(n)
Output: foodSource[·]
```

CalculateOF()

```
Input: solution[·], α1, α2, α3, α4
r_u ← 0
P_u ← 0
for each clink ∈ solution.CLinks do
    r_u ← r_u + Calculate_DataRate(clink)
    P_u ← P_u + solution.Power[clink]
end for
r_d ← 0
P_d ← 0
for each dlink ∈ solution.D2DLinks do
    r_d ← r_d + Calculate_DataRate(dlink)
    P_d ← P_d + solution.Power[dlink]
end for
OF = α1 r_u + α2 r_d − α3 P_u − α4 P_d
Output: OF
```

$O(u)$

$O(d)$

$O(u+d)$

ValidateFeasibleSolution()

```
Input: N, solution[·]
for each RB ∈ N do
    clink ← solution.GetClinkAllocation(RB)
    dlink ← solution.GetDlinkAllocation(RB)
    Check_SINR_Threshold(clik, dlink)
    Check_Transmission_Power(clik, dlink)
    Check_Communication_Mode(clik, dlink)
end for
```

$O(n)$

The computational complexity of ACS-EM is $\boldsymbol{O}(\boldsymbol{nro\_it} * \boldsymbol{ant\_quantity} * (n')^2(\boldsymbol{u}+\boldsymbol{d})^2)$

Algorithm 2. *Ant Colony System with External Memory (ACS-EM)*

```
Input: ant_quantity, α, β, q0, τ0, σ, nro_it, G_C(V_tx, Λ), u, d
Input: N_u, α1, α2, α3, α4
n ← u + N_u
τ[·] ← InitPheromone(τ0) → O((n')^2(u + d))
memory[·] ← InitMemory() → O(l(u + d))
for i = 0 to nro_it do
    for each a ∈ ant_quantity do
        currentPosition ← RootNode
        TourList ← currentPosition
        while currentPosition.Children ≠ 0 do
            η[·] ← CalculateChildrenHeuristic() → O(n')
            q ← RandomDouble()
            if q < q0 then
                r ← RandomDouble()
                if r < r0 then
                    nextPosition ← arg_max{memory[i, :]} → O(n')
                else
                    nextPosition ← arg_max{τ_ij * η_ij^β} → O(n')
                end if
            else
                z ← RandomDouble()
                if z < z0 then
                    nextPosition ← arg_min{memory[i, :]} → O(n')
                else
                    p_z[·] ← CalculateChildrenProbability(α, β, τ, σ) → O(n')
                    nextPosition ← Get_Z(p_z) → O(n')
                end if
            end if
            TourList.Add(nextPosition)
            SelectRandomPowerValue(nextPosition)
            SelectCommunicationMode(nextPosition)
            ValidateFeasibleSolution(TourList, N) → O(n)
            τ_ij = (1 − δ) · τ_ij + δ · τ0 → O((n')^2(u + d))
            currentPosition ← nextPosition
        end while
        TourList_OF ← CalculateOF(TourList, α1, α2, α3, α4) → O(u + d)
        if TourList_OF > BestTour_OF then
            BestTour ← TourList
            BetsTour_OF ← TourList_OF
        end if
    end for
    τ_ij = { (1 − δ)τ_ij + δΔτ_ij^bs,  if j ∈ Ω
           { (1 − δ)τ_ij,              if j ∉ Ω      → O((n')^2(u + d))

    memory[·] = UpdateMemory(BestTour) → O(u + d)
    if BestTour_OF > GlobalBestTour_OF then
        GlobalBestTour ← BestTour
        GlobalBestTour_OF ← BestTour_OF
    end if
end for
Output: GlobalBestTour(allocation, transmit power and communication mode)
```

$\rightarrow \boldsymbol{O}\big((\boldsymbol{u}+\boldsymbol{d})(\boldsymbol{n}'+\boldsymbol{n} + (\boldsymbol{n}')^2(\boldsymbol{u}+\boldsymbol{d}))\big) = \boldsymbol{O}((\boldsymbol{n}')^2(\boldsymbol{u}+\boldsymbol{d})^2)$

$\rightarrow \boldsymbol{O}(\boldsymbol{nro\_it} * \boldsymbol{ant\_quantity} * ((\boldsymbol{n}')^2(\boldsymbol{u}+\boldsymbol{d})^2))$

$\rightarrow \boldsymbol{O}(\boldsymbol{nro\_it} * \boldsymbol{ant\_quantity} * (\boldsymbol{n}')^2(\boldsymbol{u}+\boldsymbol{d})^2)$

InitPheromone()

```
Input: τ0
for each vertex ∈ V^u do → O(un)
    for each child ∈ vertex.Children do → O(n)
        τ[vertex, child] = τ0
    end for
end for
for each vertex ∈ V^d do → O(dn')
    for each child ∈ vertex.Children do → O(n')
        τ[vertex, child] = τ0
    end for
end for
Output: τ[·]
```

$\rightarrow \boldsymbol{O}(\boldsymbol{un}^2)$

$\rightarrow \boldsymbol{O}(\boldsymbol{d}(\boldsymbol{n}')^2)$

$\rightarrow \boldsymbol{O}(\boldsymbol{un}^2) + \boldsymbol{O}(\boldsymbol{d}(\boldsymbol{n}')^2) = \boldsymbol{O}((\boldsymbol{n}')^2(\boldsymbol{u}+\boldsymbol{d}))$ ***since*** $\boldsymbol{n}' > \boldsymbol{n}$

InitMemory()

for each $vertex \in V^u$ **do** $\rightarrow O(un)$
  $memory[vertex] \leftarrow 0$
end for
for each $vertex \in V^d$ **do** $\rightarrow O(dn')$
  $memory[vertex] \leftarrow 0$
end for
Output: $memory[\cdot]$

$\rightarrow O(un) + O(dn') = O(n'(u+d))$ *since* $n' > n$

CalculateChildrenHeuristic()

if $currentPosition \in V^u$ **then**
  for each $child \in currentPosition.Children$ **do** $\rightarrow O(n)$
    $\eta[child] \leftarrow GetHeuristicValue(child)$
  end for
end if
if $currentPosition \in V^d$ **then**
  for each $child \in currentPosition.Children$ **do** $\rightarrow O(n')$
    $\eta[child] \leftarrow GetHeuristicValue(child)$
  end for
end if
Output: $\eta[\cdot]$

$\rightarrow O(n) + O(n') = O(n')$ *since* $n' > n$

CalculateChildrenProbability()

Input: α, β, τ, η
if $currentPosition \in V^u$ **then**
  for each $child \in currentPosition.Children$ **do** $\rightarrow O(n)$
    $p_k[child] \leftarrow GetProbability(child)$
  end for
end if
if $currentPosition \in V^d$ **then**
  for each $child \in currentPosition.Children$ **do** $\rightarrow O(n')$
    $p_k[child] \leftarrow GetProbability(child)$
  end for
end if
Output: $p_k[\cdot]$

$\rightarrow O(n) + O(n') = O(n')$ *since* $n' > n$

OfflinePheromoneUpdate()

Input: δ, $BestTour$
for each $vertex \in V^u$ **do** $\rightarrow O(un)$
  for each $child \in vertex.Children$ **do** $\rightarrow O(n)$
    $\tau[vertex, child] \leftarrow GetOfflinePheromoneValue()$
  end for
end for
$\rightarrow O(un^2)$
for each $vertex \in V^d$ **do** $\rightarrow O(dn')$
  for each $child \in vertex.Children$ **do** $\rightarrow O(n')$
    $\tau[vertex, child] \leftarrow GetOfflinePheromoneValue()$
  end for
end for
$\rightarrow O(d(n')^2)$
Output: $\tau[\cdot]$

$\rightarrow O(un^2) + O(d(n')^2) = O((n')^2(u+d))$ *since* $n' > n$

UpdateMemory()

Input: $BestTour$
for each $vertex \in BestTour$ **do**
  $memory[vertex] \leftarrow memory[vertex] + 1$
end for
Output: $memory[\cdot]$

$\rightarrow O(u+d)$

The computational complexity of PSO is $O(nro\_it * particle\_quantity((u+d) * n' + n))$

Algorithm 3. *Particle Swarm Optimization (PSO)*

```
Input: particle_quantity, ω, c1, c2, nro_it, u, d
Input: N_u, α1, α2, α3, α4
n ← u + N_u → O(n'(u + d))
InitVelocity() → O(particle_quantity * (u + d))
InitPosition()
for k = 0 to nro_it do
   for each i ∈ particle_quantity do

***Velocity Updating***
      r1 ← RandomValue()
      r2 ← RandomValue()
      pbest ← Substract(BestSoFarPosition, currentPosition) → O(u + d)
      gbest ← Substract(GlobalBestPosition, currentPosition) → O(u + d)
      for each row_u ∈ V^u do
         v1 ← Multiply(c1 * r1, pbest[row_u])
         v2 ← Multiply(c2 * r2, gbest[row_u])
         v3 ← Multiply(ω, currentPosition[row_u])
         Υ^i_u ← Sum(v1, v2, v3)
      end for                                          → O(u)
      for each row_d ∈ V^d do
         v1 ← Multiply(c1 * r1, pbest[row_d])
         v2 ← Multiply(c2 * r2, gbest[row_d])
         v3 ← Multiply(ω, currentPosition[row_d])
         Υ^i_d ← Sum(v1, v2, v3)
      end for                                          → O(d)
      Υ^i ← Υ^i_u ∪ Υ^i_d

***Position Updating***
      r3 ← RandomNumber()
      for each v^i_j ∈ Υ^i do
         cut(v^i_j) ← {e | e_j p(e) ∈ Υ^i_j and p(e) ≥ r3} → O(l)
      end for                                          → O((u + d) * n')
      new_P^i ← ∅
      for each v^i_j ∈ Υ_i do
         Cand_Set^i_j ← {e | e ∈ cut(Υ^i_j)) and e satisfies constrains}
         while construction of new_P^i and Cand_Set^i_j ≠ ∅ do
            new_P^i ← SelectRandomElement(Cand_Set^i_j)
            Update(Cand_Set^i_j)
         end while                                     → O(u + d)
         if construction of new_P^i is not finished then
            Cand_Set^i_j ← {e | e ∈ P^i_j and e satisfies constrains}
            while construction of new_P^i and Cand_Set^i_j ≠ ∅ do
               new_P^i ← SelectElement(Cand_Set^i_j)
               Update(Cand_Set^i_j)
            end while                                  → O(u + d)
         end if
         if construction of new_P^i is not finished then
            Cand_Set^i_j ← {e | e ∈ E_j and e satisfies constrains}
            new_P^i ← SelectBestElement(Cand_Set^i_j) → O(n')
         end if                                        → O((u + d) * n')
      end for
      ValidateFeasibleSolution(new_P^i, N) → O(n)
      P^i ← new_P^i
      BestSoFar_OF ← CalculateOF(P^i, α1, α2, α3, α4) → O(u + d)
      if BestSoFar_OF > GlobalBestPosition_OF then
         GlobalBestPosition ← BestSoFar
         GlobalBestPosition_OF ← BestSoFar_OF
      end if
   end for                                   → O(nro_it * particle_quantity * ((u + d) * n' + n))
Output: GlobalBestPosition (allocation, transmit power and communication mode)
                                             → O(nro_it * particle_quantity * ((u + d) * n' + n))
```

InitVelocity()

```
for each vertex ∈ V^u do → O(un)
   vertex.SetInitVelocity()
end for
for each vertex ∈ V^d do → O(dn')
   vertex.SetInitVelocity()
end for
Output: velocity[·] = 0
                         → O(un) + O(dn') = O(n'(u + d)) since n' > n
```

InitPosition()

```
Input: particle_quantity
for each i ∈ particle_quantity do → O(particle_quantity)
   for each row ∈ V^u do → O(u)
      position[i].Add() ← row.Select_Random_Column()
   end for
   for each row ∈ V^d do → O(d)
      position[i].Add() ← row.Select_Random_Column()
   end for                            → O(u) + O(d)
end for                               → O(particle_quantity * (u + d))
Output: position[·]
```

C - SBRT2019 PUBLICATION.

XXXVII SIMPÓSIO BRASILEIRO DE TELECOMUNICAÇÕES E PROCESSAMENTO DE SINAIS - SBrT2019, 29/09/2019–02/10/2019, PETRÓPOLIS, RJ

Alocação de controladores SDN para gerenciamento de comunicações D2D

Yarisley Peña Llerena e Paulo Roberto L. Gondim

***Resumo*— Para emprego da rede *LTE-Advanced* (LTE-A), a comunicação "device-to-device" é uma tecnologia candidata a permitir um adequado uso de recursos, considerando perspectiva baseada na localização do usuário e em serviços baseados em proximidade. Por outro lado, redes definidas por software favorecem a melhoria da eficiência do gerenciamento e controle dos recursos de rede. Este artigo trata o problema de alocação de controladores SDN visando o adequado gerenciamento de comunicações D2D em uma rede LTE-A. Para solução desse problema, foi adotada abordagem baseada em memória externa do algoritmo de colônia de formigas, com resultados similares aos de busca exaustiva.**

***Palavras-Chave*— D2D, SDN, colonia de formigas, alocação de controladores**

***Abstract*— For the use of the LTE-Advanced network (LTE-A), device-to-device communication is a candidate technology to allow an adequate use of resources, considering user location and proximity-based services perspective. On the other hand, software-defined networks improve the efficiency of the management and control of network resources. This article addresses the problem of allocating SDN controllers for the proper management of D2D communications over an LTE-A network. To solve this problem, the ant colony algorithm based on external memory approach was adopted, with results similar to exhaustive search.**

***Keywords*— D2D, SDN, ant colony, controller allocation**

I. Introdução

O tráfego de dados móveis tem aumentado nos últimos anos e, com o advento da Internet das Coisas (IoT, do inglês *Internet of Things*) cada dispositivo será conectado à Internet, produzindo dados oriundos de processos de sensoriamento. O tráfego de dados continuará crescendo a uma taxa exponencial, o que faz com que a próxima geração de redes de telefonia móvel (5G) incorpore necessidades como a disponibilidade de vários caminhos de transferência, a diminuição da latência e alta disponibilidade, entre outros [1]. Para satisfazer os desafios mencionados estão surgindo novas tecnologias como comunicações dispositivo a dispositivo (D2D, do inglês *device-to-device*) e redes definidas por software (SDN, do inglês *software-defined networks*).

A comunicação D2D é definida como uma comunicação direta entre dois dispositivos móveis sem passar pela estação base (BS) ou pelo núcleo da rede [2]. A comunicação D2D ocorre de forma direta (par a par) na transmissão de dados, enquanto o estabelecimento e gerenciamento da conexão pode ser feito pelos dispositivos móveis ou através da infraestrutura celular. Neste trabalho é utilizada uma abordagem onde o gerenciamento da comunicação é realizado por uma entidade localizada na rede celular e que pode obter informações tanto da rede celular quanto das comunicações D2D, o que permite, por exemplo, executar escoamento de dados (*offloading*) e/ou tratar a mitigação da interferência. Considera-se que uma comunicação D2D será feita utilizando o espectro celular (*Inband*), como ocorre na tecnologia *LTE Direct*.

Por outro lado, conceitos relacionados com SDN têm sido explorados no sentido de aumentar a eficiência no gerenciamento e controle da rede. Em redes SDN o plano de controle e o plano de dados são separados, ficando toda a inteligência da rede em um ponto central chamado controlador SDN. Com isso, se consegue uma maior flexibilidade na administração das entidades de retransmissão (como eNBs - *evolved NodeBs* ou switches), permitindo que exista programabilidade o que propícia a adequação de novos serviços de rede.

Nossa proposta emprega controladores SDN, para gerenciar as comunicações D2D que acontecem na área de cobertura de cada eNB. Assumimos que os controladores são localizados numa infraestrutura celular *LTE-Advanced* (LTE-A).

O emprego de múltiplos controladores SDN na rede LTE-A leva a dois problemas: (i) obter o número de controladores necessários para gerenciar adequadamente todas as comunicações D2D e (ii) definir como realizar a alocação das eNBs aos controladores SDN. Para encontrar uma solução modelamos os problemas anteriores como um problema de otimização, onde pretendemos determinar o número mínimo de controladores SDN e a melhor atribuição balanceada de eNBs a esses controladores, além de satisfazer um tempo de resposta menor ou igual a um QoS específico.

O problema de otimização antes mencionado pode ser entendido como um problema de otimização combinatória, semelhante ao problema de "localização de facilidades - *facility location problem*", considerado um problema NP-difícil [3], e para o qual gerar uma solução ótima é uma tarefa exaustiva e demorada. Nesse cenário, as metaheurísticas representam uma alternativa para explorar o espaço de pesquisa e fornecer uma solução próxima da ótima.

Neste artigo, utilizamos uma abordagem baseada em memória externa conhecida como *Ant Colony System with external memory* (ACS-EM) [4] para encontrar o número mínimo de controladores SDN, assim como a melhor atribuição de eNB aos controladores. Investigamos o problema de alocação levando em consideração a latência de processamento nos controladores SDN dentro do tempo de resposta. Como em [1] e [5], modelamos o processamento dos controladores aplicando teoria de filas e a taxa de solicitação recebida pelos controladores SDN envolve o tráfego de controle das

Yarisley Peña, Paulo Roberto L. Gondim, Universidade de Brasília (UnB), Brasília - DF, Brasil, E-mails: yarisleyllerena@aluno.unb.br, pgondim@unb.br

IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received November 1, 2019, accepted November 19, 2019, date of publication November 22, 2019,
date of current version December 6, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2955434

SDN-Controller Placement for D2D Communications

YARISLEY PEÑA LLERENA, (Member, IEEE),
AND PAULO R. L. GONDIM, (Senior Member, IEEE)
Department of Electrical Engineering, Brasília University (UnB), Brasília 70910-900, Brazil
Corresponding author: Yarisley Peña Llerena (yarisleyllerena@aluno.unb.br)

This work was supported in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES).

ABSTRACT The increasing demand for data transmission resources to be handled by next generation cellular networks has led to the emergence of new technologies, such as Device-to-Device (D2D) communications and software-defined networks (SDN). D2D expands the use of resources from a location perspective and SDN enables an efficient management and control of the available resources. This article addresses a study of the influence of D2D communications management on a Long Term Evolution (LTE) network with SDN controllers regarding the obtaining of their necessary number and best location within the cellular infrastructure. The controller placement problem was modeled as an optimization problem and solved by the ant colony system with external memory (ACS-EM) algorithm. The proposed algorithm was compared with a particle swarm optimization (PSO)-based algorithm and its effectiveness has been validated.

INDEX TERMS Ant colony system with external memory, controller placement problem, D2D, software defined networking.

I. INTRODUCTION

Cellular networks have evolved according to the population's needs. As such, new forms of evolution in telecommunications have changed the current paradigms.

Fifth-generation communications or 5G Systems have been designed to offer both more advanced and complex configurations aimed at better performance to meet the requirements of society and a new form in the thinking and components that comprise them. Such a technology will provide unlimited access to information, as well as availability for its sharing anywhere, at any time, by any person or thing, for the benefit of society [1].

5G communications are characterized by numerous devices and interconnected networks and an increase in the data traffic in comparison to the current one. Other characteristics include seamless integration of heterogeneous networks, use of femtocells and increased capacity and performance with reduced latency [2], [3]. Moving networks, ultra-dense networks, device to device (D2D) communications, ultra-reliable communications, and mass communication of machines have also been considered.

The associate editor coordinating the review of this manuscript and approving it for publication was Antonino Orsino.

D2D communication refers to the connection among devices through any network infrastructure, controlling radio access resources and direct links for minimizing the resulting interference. It also provides an alternative mode of communication that increases connectivity, utilization of the spectrum, and coverage area [3].

D2D communication is defined as a direct communication between two mobile devices without passing through a cellular base station (BS) or network core [4]. It requires different levels of control by the operator. Based on the business model, it has either full, or partial control over the resource allocation among source, destination, and relay devices, or no control. Below are the different types of D2D communications defined according to the function of the cellular infrastructure (Bastos and Cecílio Da [5]):

1) D2D communication with establishment of an operator-controlled connection: source and destination devices talk and exchange data with each other with no BS; however, they are served by a BS to establish the link.
2) D2D communication with the establishment of a device-controlled connection: source and destination devices communicate directly with each other with no operator control, and must implement methods that

E - COMPUTER NETWORKS PUBLICATION.

Computer Networks 183 (2020) 107581

Contents lists available at ScienceDirect

Computer Networks

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

Social-aware spectrum sharing for D2D communication by artificial bee colony optimization

Yarisley Peña Llerena *, Paulo R.L. Gondim

Department of Electrical Engineering, Brasilia University (UnB), Brasília, Brazil

ARTICLE INFO

Keywords:
Social relationship
D2D
RRA
ABC
Partial CSI
Feedback

ABSTRACT

Next-generation cellular networks must handle an increasing demand for data transmission resources, which requires the study of new technologies, such as Device-to-Device (D2D) communications. D2D expands the use of resources from a location perspective, and takes advantage of the geographical proximity of users to perform data transmission at high rates and low propagation delays. On the other hand, the widespread use of social networks establishes the degree of social relations among users and facilitates the efficient management of D2D communication networks. The integration of social networks and D2D communications into 5G networks has posed important challenges (e.g., exploitation of the social relationships of mobile users and management of interference and resources towards a better performance of D2D communications). This article proposes a social-aware Radio Resource Allocation (RRA) optimization solution for D2D communications in 5G networks that considers partial Channel State Information (CSI). A selected-NM Maximum Distance Ratio (MDR) q-bit feedback scheme designed reduces feedback overhead, since each D2D receiver sends only q-bit feedback CSI of N cellular User Equipment (CUE) and M D2D pairs with the largest MDR metric. The RRA problem is solved by an Artificial Bee Colony (ABC) algorithm for the obtaining of number of D2D connections in each evolved NodeB (eNB) and maximization of the system throughput.

1. Introduction

The rapid development of smart devices has led to an exponential growth in the demand for services with high data consumption. Applications with high data transmission rates have become increasingly prevalent, thus imposing a new traffic pressure on the cellular network. Therefore, the next generation of mobile technologies is expected to connect people, things, data, applications, transport systems, and cities in smart networked communication environments; it should transport a considerable amount of data much faster, reliably connect a huge number of devices, and process very high volumes of data with minimal delay [1].

Wireless networks have evolved to a new generation (5G) through the optimization of Long Term Evolution (LTE) and LTE-A for satisfying the increasing demand for data transmission. Interest has been devoted to the study of techniques that enable communication among devices through different access networks and consider the possibilities offered by the geographic neighborhood of a communication pair. Device-to-Device (D2D) communication has shown a promising technology for the achievement of those objectives.

D2D communication is defined as the direct communication between two mobile devices without passing through a cellular Base Station (BS) or network core [2]. Its important applications in 5G [3,4] comprehend extension of the coverage of mobile networks and offer of multiple services based on the proximity of the devices (e.g., local information exchange, multimedia traffic in Internet of Things (IoT) systems, gaming, social applications, vehicle-to-vehicle communications in Intelligent Traffic Systems (ITS), among others). Recent research [3] has focused on the benefits of D2D communications, such as higher data rate, low delay, coverage extension, reliability in communications, even in network failures, low energy consumption, traffic offloading, and possible instantaneous communications between devices, for the meeting of requirements imposed by 5G networks.

Moreover, the widespread dissemination of social networks (as Facebook, Whatsapp, Instagram, Twitter, among others) has required adequate collect and processing of the demand by each mobile telecom operator. The possible use of control information on such social network information enables devices of some computational capacity to identify interesting or related peers for the exchange of information and context-based services, among other options. The wide adoption of social networks has given rise to a new dimension in the design of

* Corresponding author.
E-mail addresses: yarisleyllerena@aluno.unb.br (Y.P. Llerena), pgondim@unb.br (P.R.L. Gondim).

https://doi.org/10.1016/j.comnet.2020.107581
Received 2 March 2020; Received in revised form 13 August 2020; Accepted 27 September 2020
Available online 6 October 2020
1389-1286/

Resource allocation based on bio-inspired algorithms for D2D cellular networks with URLLC services

Yarisley Peña Llerena, *Member, IEEE,* and Paulo R. L. Gondim, *Senior Member, IEEE*

***Abstract*—5G networks promise faster data transmission, higher bandwidth, and lower latency. Device-to-Device (D2D) communication is a technology that contributes to 5G becoming a reality, considering proximity services (ProSe) of communication devices. This article addresses the use of D2D for Ultra-reliable and Low Latency Communication (URLLC) in a wireless Internet of Things (IoT) industrial environment and presents the formulation of a radio resource allocation problem for the definition of the number of D2D communications allowed in the network towards maximizing of its energy efficiency. The problem is solved by three bioinspired algorithms that take into account interference when cellular and D2D users use the same radio resource.**

***Index Terms*—URLLC, D2D, RRA, ABC, PSO, ant colony, resource allocation, bio-inspired.**

I. Introduction

DEMANDS for data consumption in cellular networks have substantially increased, and the spread of applications that require higher-quality services with low latency has led to the evolution of such networks. 5th generation (5G) cellular networks must support a number of emerging applications with varying Quality of Service (QoS) requirements, leading to services categorized as Enhanced Mobile Broadband (eMBB), Massive Machine Type Communication (mMTC), and Ultra-reliable and Low Latency Communication (URLLC) [1].

Industry 4.0 is expected to integrate organizations, people, and assets combining intelligent technologies with advanced production and operation techniques. New technologies such as artificial intelligence, robotics, cognitive technologies, cloud computing, Internet of Things (IoT), big data, among others must emerge or efficiently developed. 5G wireless technologies are expected to improve performance towards mission-critical applications (e.g., factory automation and control for Factories of the Future, autonomous vehicles, smart grid automation, unmanned aerial vehicles control information delivery, and remote surgery [2], [3]). Such applications have strict requirements in terms of latency and reliability, which can be met by URLLC services, since they support real-time, ultra-reliable, and low latency communications [1], [2], [4].

In URLLC communications, the decoding error probability should be lower than 10^{-5} and no higher than 1 ms latency for a small packet size (e.g., 32 bytes) [5]. Ultra-high reliability and low latency are the major technical challenges in URLLC networks [6]–[11], with a tradeoff between them. Specifically, reliability can be improved by techniques such as signaling and relaying, which increase latency [6]. On the other hand, some wireless networks focus on long packet transmission scenarios to maximize system throughput and favor energy efficiency. The simultaneous assurance of high reliability and low latency is challenging, which highlights the need for new technologies that meet URLLC service requirements.

As a traditional metric in information theory, Shannon capacity has been widely used to characterize the maximum achievable transmission rate [6]–[8], considering channels with Additive White Gaussian Noise (AWGN). Channel capacity bounds determine the maximum data rates to be transmitted with low error probability, assuming no constraints on delay and encoder and decoder complexity. However, in scenarios which require the use of short packets, the impact of packet size and decoding error is more prominent and cannot be ignored in the aforementioned channel capacity metric [9], [12].

Device-to-Device (D2D) communication enables physically nearby User Equipment (UE) to communicate directly, with no evolved NodeB (eNB) intervening in data traffic. Consequently, the latency between two neighboring devices can be highly reduced through the use of a D2D proximity link, guaranteeing the required data rate. The integration of D2D into future industrial wireless networks and next-generation manufacturing can support massive machine-like connections. A D2D communication system can satisfy URLLC service requirements, as well as improve spectrum efficiency, energy efficiency, and overall network capacity.

This paper deals with the resource allocation problem in a D2D cellular network that uses URLLC services to send short packets directly to their destination. Since such packets are essential for low latency critical mission communications, the channel coding rate is adopted as a performance metric in D2D links in the finite packet length regime, as defined in [13]. It provides the maximum number of data bits sent in a packet of size m and has been used as a performance metric in systems with URLLC services [6]–[11].

Resource blocks are assigned to UEs and the QoS of both cellular User Equipment (CUE) and D2D-capable User Equipment (DUE) is considered. Mutual interference between different D2D peers and between DUEs and CUE that share the same resource block (RB) are taken into account. This

Yarisley Peña Llerena was with the Department of Electrical Engineering, Brasilia University (UnB), Brasília, Brazil. E-mail: yarisleyllerena@aluno.unb.br

Paulo R. L. Gondim was with the Department of Electrical Engineering, Brasilia University (UnB), Brasília, Brazil. E-mail: pgondim@unb.br

Manuscript received April 19, 2021; revised August 16, 2021.

Energy-efficient optimization for URLLC services in D2D cellular networks

Yarisley Peña Llerena
Department of Electrical Engineering
Brasilia University (UnB)
Brasília, Brazil
yarisleyllerena@aluno.unb.br

Paulo R. L. Gondim
Department of Electrical Engineering
Brasilia University (UnB)
Brasília, Brazil
pgondim@unb.br

***Abstract*—5th generation (5G) networks consider an architecture that will support the main scenarios identified in wireless networks. Such scenarios are classified into three categories of services, of which one is Ultra-reliable and Low Latency Communication (URLLC), which must ensure high reliability and low latency. This paper focuses on the resource allocation considering URLLC services in a Device-to-Device (D2D) cellular network. The joint problem of mode selection and power and channel allocations is investigated and an optimization problem is formulated, aiming to maximize the energy efficiency according to cellular (traditional) and URLLC services. A bioinspired algorithm is proposed for solving the problem and simulations performed showed it can generate good solutions to improve the network energy efficiency.**

***Index Terms*—URLLC, D2D , RRA, ABC.**

I. INTRODUCTION

DEMANDS for data consumption in cellular networks have substantially enhanced; consequently, applications that involve better-quality services with low latency have led to the evolution of such networks. Several new applications with varying Quality of Service (QoS) requirements must be sustained by 5G cellular networks, which has led to services categorized as Enhanced Mobile Broadband (eMBB), Massive Machine Type Communication (mMTC), and Ultra-reliable and Low Latency Communication (URLLC) [1].

Low latency and ultra-high reliability are ambitious requirements in URLLC networks [2]–[6], with a tradeoff between them. Specifically, reliability can be improved by techniques such as signaling and relaying, which increase latency [2].

On the other hand, some wireless networks focus on long packet transmission scenarios towards maximizing system throughput and favoring energy efficiency. The simultaneous assurance of high reliability and reduced latency is challenging, which highlights the need for new technologies that meet URLLC service demands. A Device-to-Device (D2D) communication system can satisfy them and improve spectrum efficiency, energy efficiency, and overall network capacity.

This paper deals with the Radio Resource Allocation (RRA) strategies in a D2D cellular network that employs URLLC services to send short packets directly to their destination. Since such packets are essential for low latency critical mission communications, the channel coding rate is adopted as a performance metric in D2D links in the finite packet length regime, as defined in [7]. It provides the maximum number of data bits sent in a packet of size m and has been used as a performance metric in systems that involve URLLC services [2]–[6].

A scenario where cellular User Equipment (CUE) uses conventional network services and D2D-capable User Equipment (DUE) uses URLLC services is adopted and the QoS of both DUE and CUE and mutual interference between different D2D peers and between DUEs and CUE sharing an identical resource block (RB) are considered. The objective of the RRA problem is the maximization of the network energy efficiency (η_{EE}). The main contribution is based on a bioinspired algorithm, known as Artificial Bee Colony (ABC), whose objective is an adequate quality RRA solution in reasonable time, since RRA has been found to be NP-hard [8]. Our solution is compared with an exhaustive search solution, taking into account metrics such as number of D2D communications and system throughput.

The structure of the manuscript is as follows: Section II addresses and briefly compares related work; Section III presents the system and interference models; in Section IV, the problem of radio resource allocation is formulated, and an innovative optimization strategy based on artificial bee colony strategy is introduced; Section V reports the experimental results and an analysis of the algorithm; finally, Section VI concludes the study.

II. RELATED WORK

In URLLC services, the block length of the channel codes must be short towards reductions in the transmission delay; the communication is no longer arbitrarily reliable and the decoding error probability should not be negligible.

Shannon's capacity is not applicable for the characterization of the maximum achievable rate of short URLLC packets, as shown by Y. Polyanskiy et al. [7], who derived a fundamental association among achievable rate, packet length and decoding error probability. Their results have been widely used in such

This study was partially financed by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), Brasil (Finance Code 001) (a scholarship was awarded to Yarisley Peña Llerena). The authors acknowledge the University of Brasilia, specially the Post-Graduation Program in Electrical Engineering (PPGEE), for the research support.

[ICIN 2023] Your paper #1570870093 ('Energy-efficient optimization for URLLC services in D2D cellular networks')

1 mensaje

ICIN 2023 <icin2023-chairs@edas.info> 11 de enero de 2023, 11:19

Para: Yarisley Llerena <yarisleyllerena@aluno.unb.br>, Paulo Roberto de Lira Gondim <pgondim@unb.br>

Paper #: 1570870093
Title: Energy-efficient optimization for URLLC services in D2D cellular networks
Author(s): Yarisley Llerena and Paulo Roberto de Lira Gondim

Dear Mrs. Llerena:

We are pleased to inform you that your paper #1570870093 has been accepted for presentation at the 2023 26th Conference on Innovation in Clouds, Internet and Networks and Workshops (ICIN) - Full papers track, to be held in Paris, France on March 6-9, 2023 (See https://www.icin-conference.org/).

ICIN 2023 implemented a thorough review process coordinated by the Technical Program Committee (TPC) Chairs. Each submitted paper received at least 3 reviews provided by TPC members or other expert reviewers. The final acceptance rate for full papers is about 38%.

The reviews of your paper can be found at https://edas.info/showPaper.php?m=1570870093 and are listed below. We kindly remind you to take the reviewers comments carefully into consideration as you prepare the final version of your paper. The TPC co-chairs may reject final papers that do not take into account reviewers' comments and expectations in an acceptable manner.

Please submit the camera-ready paper in PDF format to the EDAS system by January 30, 2023 adhering to the IEEE double-column format, and IEEE eXplore PDF-compliant.
Detailed camera-ready author instructions are available on the website here: https://www.icin-conference.org/camera-ready/

Also, please select the name of the presenter in the EDAS paper page as soon as this information is available. It is mandatory that at least one of the authors of your paper registers at the AUTHOR rate.
Please note that only papers presented at the conference by one paper co-author will be submitted to IEEE Xplore. Registration information will be posted soon at https://www.icin-conference.org/

The technical program of the conference including the presentation schedule will be announced and published soon.

Once again, we express our congratulations on the acceptance of your paper. As we are preparing a very attractive program, we are very much looking forward to your participation at ICIN 2023.

Best regards,

Walter Cerroni, Mika Ylianttila
ICIN 2023 TPC Chairs

Diego Lopez, Marie-José Montpetit
ICIN 2023 General Chairs