IADIS International Journal on Computer Science and Information Systems
Vol. 4, No. 3, pp. 111-122
ISSN: 1646-3692

STUDIES ON THE COMPUTATIONAL SCALE OF A DISTRIBUTED RCA ALGORITHM

Xiaoguang Ma. *Department of Electrical & Computer Engineering. Florida State University. 2525 Pottsdamer Street, Tallahassee , FL 32310 USA*

Ming Yu. *Department of Electrical & Computer Engineering. Florida State University. 2525 Pottsdamer Street, Tallahassee , FL 32310 USA*

Bing W. Kwan. *Department of Electrical & Computer Engineering， Florida State University. 2525 Pottsdamer Street, Tallahassee , FL 32310 USA*

ABSTRACT

For the Radio Channel Allocation (RCA) of wireless networks, how to efficiently allocate the limited number of channels to achieve high throughput is a challenging problem. The major difficulty in solving the RCA problem is to maximize the throughput of the entire network using the min-max optimization scheme. Usually, it is solved by using various heuristic methods, which are known to be NP-hard and have unknown computational scales. In this paper, we analyze a typical RCA algorithm for IEEE 802.11 based wireless networks including wireless LANs (WLANs) and wireless mesh networks, namely, the distributed heuristic algorithm (DHA) [2], by using both analytical and statistical analysis in terms of the computational scale (CS) of the method. The CS of an algorithm is defined as the number of channel reallocation times until the network reaches a convergence state. By extensive simulations, we demonstrate that DHA reaches the convergence state in finite steps. The total number of channel reallocations is a log-logistic distribution. Based on all the possible network configurations, we develop a method to estimate the CS. We find that the overall upper limit of the CS for a network is O(I), where I is the number of access points (APs) or mesh routers that are responsible for allocating the available radio channels.

KEYWORDS

Radio Channel Allocation, Distributed Heuristic Algorithm, Computational Scale.

1. INTRODUCTION

The RCA of the IEEE 802.11 WLANs with multiple access points (APs) is an important problem since the standard only provides a limited number of radio channels. How to allocate the limited number of channels dynamically and efficiently has not been well investigated. Usually, the RCA problem is formulated as a min-max optimization problem with respect to channel utilization [1], [2]. The difficulty in solving the min-max optimization problem is that it is NP-hard and has no analytical solution. Therefore, various heuristic algorithms have been proposed, including centralized [3] and distributed algorithms [4].

It is known that channel allocations of a WLAN are impacted by many factors, the two most important of which are the number of APs in the network and the number of stations associated with an AP. The latter is normally called the load of an AP. For a centralized heuristic algorithm (CHA), a network needs to have an overall control center to conduct channel allocations. The channel allocation and reallocation decisions are made by the control center based on the load information gathered by the control center for each AP. For a distributive heuristic algorithm (DHA), the network does not rely on a control center to allocate channels. Each AP collects its own load information and makes channel reallocation decisions independently.

In this paper, we compare the DHA to the CHA in [6] and prove that the DHA performs equally well or even better than the CHA in the scenarios considered in [6]. We further study the computational scale of the DHA. A theoretical analysis is given and compared to the simulation results. Estimates of other cases are extrapolated by using a statistical method and by sorting the network settings into several different groups. Finally we present our overall studies on the computational scales of the DHA.

2. DHA ALGORITHMS

For a WLAN, the coverage area of the network is divided into I sub-areas. For simulation purpose, the I sub-areas are arranged in a matrix of N rows by N columns, where $I = N$xN. Each sub-area is a square with an AP placed at the center of square. Thus there are I APs in the network. We assume that there are J non-overlapping channels, indexed by j, $j \in I_J = \{1, 2 . . ., J\}$ available for the APs and stations in the network. For example, $J = 3$, for the IEEE 802.11g standard based networks. We also assume that there are M stations uniformly deployed in each sub-area. Each station is formally associated with only one AP at the center of the square and thus assigned with one radio channel by the AP.

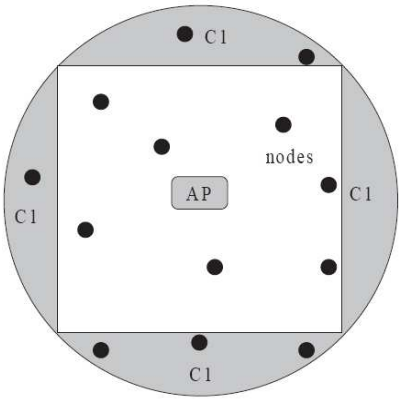

Figure 1. The Coverage Area of an AP

The interference region within an AP is defined as the area between its radio range and a co-located square area within the radio range, as shown in Fig. 1. The interference region, denoted by C1, consists of four parts that belong to four immediate neighbors of the central AP. The transmission of any station inside C1, if using the same channel as the central AP, will cause enough interference to defer the AP's transmitter or cause collision in the AP's receiver.

The load of the AP i in channel j is denoted by L_i^j. It is comprised of two parts: the number of stations formally associated with the AP (in the square area) plus the number of stations in the interference region that share the channel j with the AP. The RCA goal is to effectively allocate J non-overlapping channels to I APs in a distributed fashion such that the following objective function can be achieved:

$$\min_j \max_i \{L_i^j\}, \qquad \text{for } i \in I_I, \quad j \in I_J$$

The objective function is to assign the channels such that the load of the most heavily loaded AP is minimized. The min-max operation will result in more resources available for the most heavily loaded AP during each channel allocation period. As the channel is dynamically allocated and reallocated, the overall network throughput will be improved.

The algorithm should ensure that the following two conditions are satisfied:

1） The traffic demand for an AP from all its associated stations should be less than the maximum data rate the AP can provide.

2） All the traffic can be sent eventually. Therefore the channels are not jammed all the time.

Since the RCA is a global optimization problem, it is usually formulated as an ILP problem, which is found to be NP-complete or NP-hard. To avoid the NP-hard problem, we develop distributive heuristic algorithms to find a suboptimal solution locally, namely we seek to find:

$$\min_j \{L_i^j\}, \qquad \text{for each } i \in I_I, \quad j \in I_J,$$

where it is assumed that each AP first operates on its maximum throughput by choosing an optimal window size in terms of the its M value [2]. Then, each AP runs the DHA periodically

and independently. It periodically scans all available channels and maintains a record for each channel. It will then stay at the channel that is found to be locally optimal.

2.1 DHA Outline

The DHA can be summarized as follows [2].

1) Randomly assign each AP one of the available channels.
2) Based on its current channel assignment, each AP calculates its load. The load of AP_i is denoted by n_i.
3) Each AP_i computes the loads of operating on the other two alternative channels. Select the minimum load and the corresponding channel number, denoted by m_i and Ch_i, respectively.
4) If $m_i < n_i$, AP_i switches its channel to Ch_i.
5) Apply Step 3 and Step 4 when an AP senses any change of channel among its neighbors. It can also periodically reallocate the channels for changing traffic load.

It is noted that we do not consider the possibility that an AP reallocates to an equal loaded channel in Step 4. This is a simplification to the problem without losing generality.

The pseudocode of the DHA is listed below:

Part I: Initialization
Comments: There are I APs in the network (*I* = NxN, i.e., N columns by N rows)
for {*i* from 1 to *I*}
do {
Randomly assign AP_i a channel.
Randomly assign the neighbors of AP_i an interference relationship vector.
}
Part II: Main
Comments: There are *J* non-overlapping channels available for allocation.
Load(i,j) := the load of AP_i when it operates at channel *j*.
while {At least one AP changes its Channel}
do {
for {*i* from 1 to *I*}
do {
for {*j* from 1 to *J*}
do {
Calculate *Load(i,j)*
}
Allocate AP_i channel *j*, which minimizes *Load(i,j)* among *j*s.
}
}
Part III: Dynamic Adaption
Comments: Δ *Load(i,j)* := the change of the load of AP_i since the last channel allocation.
α_0:= a predefined constant that indicates the QOS requirement of the network.
Periodically calculate the *Load(i,j)*
while { Δ *Load(i,j*) $\geqslant \alpha_0$}
do {

Repeat the Main progress.
}
Part IV: End.

2.2 Network Relationship Model

In this section, we reveal the essential relationship between an AP and its neighbors by assigning interference coefficients to its neighbors. The network has I APs arranged in an NxN matrix. In this way, any AP has four neighbors, except for those APs in the network edge. Suppose the four neighbors of one AP have interference factors that can be ranked in ascending order as: $a \leqslant b \leqslant c \leqslant d$, as shown in Fig 2. These coefficients just indicate the strength of the influence of the neighbors and do not need to be exactly the number of nodes that fall into the corresponding regions.

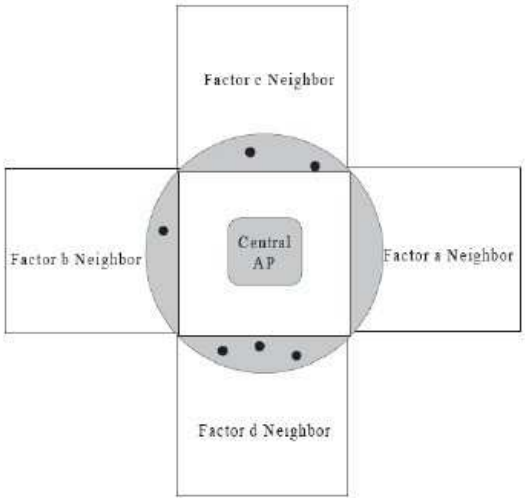

Figure 2. The neighboring APs of an AP.

We can model the one-one (e.g. a v.s. b), and two-one (e.g. a+b v.s. c) relationship between the central AP's four neighbors to be 50 possible combinations. We do not consider two-two relationship (e.g. a+c v.s. b+d) and three-one relationship (e.g. a+b+c v.s. d) since those relationships will not affect the way the central AP reallocating its channel. For example, if we have c < a+b < d, then the central AP will reallocate its channel to share the channel with the neighbor that has the interference factor "c". However, suppose we have neighbors "a" and "c" share one channel while "b" and "d" share another, whether a + c greater than, equal to or less than b + d, the central AP will switch to the third channel.

All the possible combinations are shown in Table 1. The first column shows the sequence numbers of the neighbor relationships. The second to the eighth columns show the relationships between the four neighbors. Note that "g" means greater than, "e" means equal, and "l" means less than. For the purpose of computer simulation, we need to digitize the relationships. We assign *a*, *b*, *c*, *d* each an integer value according to their relationships specified by the previous 7 columns. The last four columns show the digitized result.

Table 1. Digital 1-1/1-2 neighbor relationship table

	a vs b	b vs c	c vs d	a+b vs c	a+b vs d	a+c vs d	b+c vs d	a	b	c	d		a vs b	b vs c	c vs d	a+b vs c	a+b vs d	a+c vs d	b+c vs d	a	b	c	d
1	e	e	e	g	g	g	g	1	1	1	1	26	e	l	l	e	l	l	l	1	1	2	4
2	l	e	e	e	e	e	g	0	1	1	1	27				e	l	e	e	1	1	2	3
3				g	g	g	g	1	2	2	2	28				e	l	g	g	2	2	4	5
4	e	l	e	l	l	e	e	0	0	1	1	29				g	l	l	l	2	2	3	6
5				l	l	g	g	2	2	5	5	30				g	l	e	e	2	2	3	5
6				e	e	g	g	1	1	2	2	31				g	l	g	g	3	3	5	7
7				g	g	g	g	2	2	3	3	32				g	e	g	g	2	2	3	4
8	e	e	l	e	l	l	l	0	0	0	1	33				g	g	g	g	3	3	4	5
9				g	e	e	e	1	1	1	2	34	l	l	l	l	l	l	l	1	3	5	9
10				g	l	l	l	1	1	1	4	35				l	l	l	e	1	3	5	8
11	l	l	e	l	l	e	g	0	1	2	2	36				l	l	l	g	1	3	5	7
12				l	l	g	g	1	2	4	4	37				l	l	e	g	1	3	5	6
13				e	e	g	g	1	2	3	3	38				l	l	g	g	2	3	6	7
14				g	g	g	g	2	3	4	4	39				e	l	l	l	1	3	4	8
15	l	e	l	e	l	l	l	0	1	1	3	40				e	l	l	e	1	3	4	7
16				e	l	l	e	0	1	1	2	41				e	l	l	g	1	3	4	6
17				g	l	l	l	1	2	2	5	42				e	l	e	g	1	3	4	5
18				g	l	l	e	1	2	2	4	43				e	l	g	g	2	3	5	6
19				g	l	l	g	2	4	4	7	44				g	l	l	l	2	3	4	8
20				g	e	e	g	1	2	2	3	45				g	l	l	e	2	3	4	7
21				g	g	g	g	2	3	3	4	46				g	l	l	g	2	3	4	6
22	e	l	l	l	l	l	l	0	0	1	2	47				g	l	e	g	3	4	5	8
23				l	l	l	e	1	1	3	5	48				g	l	g	g	3	4	6	8
24				l	l	e	e	1	1	3	4	49				g	e	g	g	2	3	4	5
25				l	l	g	g	2	2	6	7	50				g	g	g	g	3	4	5	6

It is natural that there is more than one way to represent a single relationship with integer vectors. For example, the 8th row represents the relationships: $a = b$, $b = c$, $c < d$, $a + b = c$, $a + b < d$, $a + c < d$, $b + c < d$. This row may be digitized to 0, 0, 0, 1 or 0, 0, 0, 2. To simplify the digitization result, we select the smallest integer vector, which is 0, 0, 0, 1. Similarly, the 9th row represents the relationships: $a = b$, $b = c$, $c < d$, $a + b > c$, $a + b = d$, $a + c = d$, $b + c = d$. Clearly, it may be digitized to 1, 1, 1, 2 or 2, 2, 2, 4. We once again select the smallest vector 1, 1, 1, 2 among all the candidate vectors.

2.3 Computational Scale Estimation based on the Central Limit Theorem

In this section, we make an attempt to analyze the simplest relationship between the neighbors resulting in the analytical probability of the upper limit of the computational scale. The focus is placed on case No. 8.

The simplest network setting of case No. 8 has four neighbor factors $a = b = c = 0$, $d = 1$. In this case the central AP's channel selection is only affected by the one neighbor with relation factor 1. The relationships between the APs can be sorted into three basic classes as follows.

1) Paired: two APs have the relation factor "1" on each other.

2) Dependent on Paired: the AP's factor "1" is on a paired AP.

3) Other.

These three classes are examined one by one in the following.

For the first class, two APs have the relation factor "1" on each other. Since we do not consider the APs on the border and each AP has four neighbors, then P(AP is one of the paired APs) = 1/4 .

Initially, every AP is randomly assigned a channel from the three independent channels (1, 6, 11), namely R, G, B. Then every AP searches all the three channels and stays with its original channel with a probability denoted by $P(0|I)$, meaning 0 change under the initial condition. Accordingly, it changes its channel with a probability $P(1|I)$.

Since the number of channels is three, the two paired APs share the same channel with probability 1/3. Only one of them needs channel reallocation.

We have $P(1|I, class\ 1) = 1/3 \times 1/2 = 1/6$ and $P(0|I, class\ 1) = 1 - P(1|I, class\ 1) = 5/6$. There will not be more than one channel reallocation within the pair.

For the second class, central AP's neighbor of factor "1" is one of the paired APs. We should exclude the paired APs, so we select from the rest 3/4 APs. The probability of having a paired AP as the factor "1" neighbor is 1/4. Thus the proportion of the second class APs is calculated by $1/4 \times 3/4 = 3/16$.

In this class, $P(0|I, class\ 2) = 2/3$ and $P(1|I, class\ 2) = 1/3$.

Besides the reallocation under the initial condition, if one AP's neighbor of factor "1" reallocates its channel (with probability $P(1|I, class\ 1)$) to share the same channel with the central AP (with probability 1/2, for reallocation, only two other channels are available), the central AP has to reallocate (again). The probability of that situation, defined as the reallocation due to the neighbor of factor "1"'s reallocation, is calculated by $P(1|NR, class\ 2) = P(1|I, class\ 1) \times 1/2 = 1/12$.

Then we can calculate the probability of 0/1/2 channel reallocation(s) of one AP in class 2.

1) $P(0|class\ 2) = P(0|I, class\ 2) \times (1 - P(1|NR, class\ 2)) = 2/3 \times (1 - 1/12) = 11/18$.

2) $P(1|class\ 2) = P(1|I, class\ 2) \times (1 - P(1|NR, class\ 2)) + P(0|I, class\ 2) \times P(1|NR, class\ 2) = 13/36$.

3) $P(2|class\ 2) = P(1|I, class\ 2) \times P(1|NR, class\ 2) = 1/36$.

For the third class, we have $P(0|I, class\ 3) = 2/3$ and $P(1|I, class\ 3) = 1/3$ as in the second class, while $P(1|NR, class\ 3) = 1/2 \times P(1|I, class\ 3) = 1/6$.

Since the probability is very small for an AP to perform three or more channel reallocations in this sparse network setting, we assume there are no more than three reallocations for one AP.

We summarize and enter the results in Table 2.

Table 2. Neighbor relationship case 8 study table

Number of Channel Reallocations	Class 1 P=1/4	Class 2 P=3/16	Class 3 P=9/16	Total probability	Simulation result
0	5/6	11/18	5/9	0.635	0.6436
1	1/6	13/36	7/18	0.328	0.3208
2	0	1/36	1/18	0.0365	0.0340

The Central Limit Theorem enables us to estimate the upper limit of the computational scale of a network purely tied together by the relationships encompassed by case No. 8. The average number of channel allocation of one AP is calculated by

$$\overline{X} = \sum_{i=0}^{2} (X_i \times P(X_i)) = 0.401$$

where $X_i = 0,1,2$. The variance is obtained as

$$\sigma = \sqrt{\sum_{i=0}^{2} (X_i - \overline{X})^2 \times P(X_i)} = 0.5596$$

For the network of I APs, the average computational scale may be approximated by a normal distribution $\mathcal{N}(0.401, 0.5596/\sqrt{I})$. The upper limit of the 95% confidence interval (CI) is $(0.401I + 1.96\sigma \times \sqrt{I})$.

To verify this analysis, we simulate with a network with 50 × 50 APs. The neighbor relation vector is selected to be [0 0 0 1]. To remove the effects of the border, we choose the lines from 11 to 40, thereby resulting in 900 APs being included. We fit the simulation results to a normal distribution model as shown in Fig. 3.

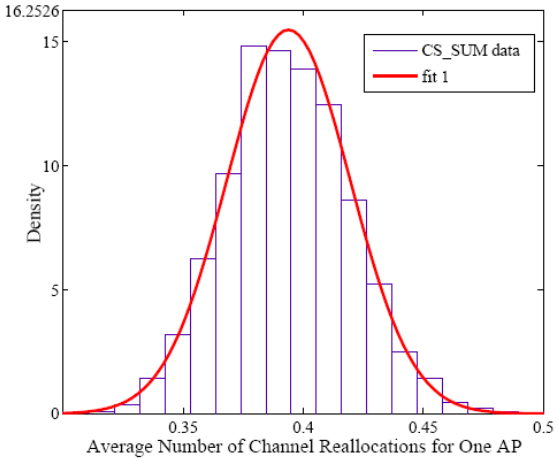

Figure 3. Normal fit of the computation scale of case No. 8

The fitted values are shown in Table 3. It can be seen that the simulation results are in exceedingly good agreement with the analytical results. The minor error is mainly due to the

assumption that one AP can take two channel allocations at most. The error does not significantly affect our results and therefore can be neglected.

Table 3. Statistical fit results of the average computational scale of case No. 8

Parameter	Estimate	Std. Err.	Analy. Result
μ	0.393866	0.000257736	0.401
σ	0.0257736	0.000182261	0.0187

It is extremely difficult to analyze all 50 relationships. Therefore, we study the property of other relationships by simulation only. The simulation results are discussed in the Appendix. Notice that the maximum mean and variance are reached in the last group with a < b < c < c, a + b > c, where μ = 0.7483 and σ = 0.0326.

However, it is recognized that a simple network composed of only one relationship does not exist. We simulate 35 groups of network settings. For each AP, we randomly choose one relationship vector out of the fifty relationships. To remove the border effects, which decrease the average mean and variance of the computational scale, we only count the 36 nodes in the center (namely rows 3 to 8 and columns 3 to 8). The box plots of the computational scale of the thirty five groups are shown in Fig. 4.

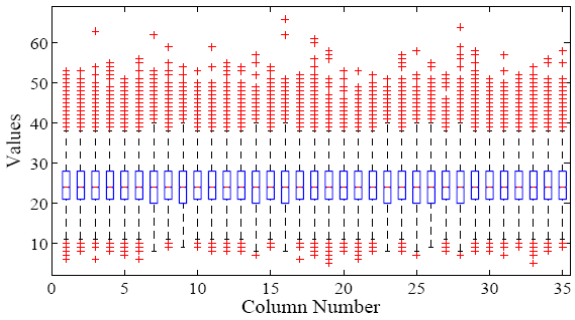

Figure 4. The box plot of total computational scale of 35 groups

We use the one-way ANOVA (analysis of variance) statistics tool in MATLAB to verify that different initial neighbor relationships of the APs will not affect the distribution of the computational scales. One-way analysis of variance (ANOVA) tests allow the determination if one given factor, such as the initial neighbor relationships of the APs, has a significant effect on the results, such as the computational scales. A *p-value* resulting from a one-way ANOVA test indicates whether the testing factor is significant or not. The ANOVA test enables all groups to be compared with each other simultaneously rather than individually by the *t*-tests.

The null hypothesis of the ANOVA test states that there are no differences between means of different groups. In the case under consideration, we consider the null hypothesis to be that the computational scales of all the groups are from the same distribution. Since we choose the number of instances to be a very large number, the random errors are supposed to follow a normal distribution and hence satisfy the one-way ANOVA assumption.

The Matlab ANOVA function produces the results tabulated in Table 4. The contents of each column in the table are described as follows:

1) The first column shows the source of the variability.

2) The second column shows the sum of squares (SS) due to each source. Rows signify the pertinent quantities between groups, whereas Error refers to the pertinent quantities within groups.
3) The third column shows the degrees of freedom (df) associated with each source.
4) The fourth column shows the mean squares (MS) for each source, which is equal to the ratio SS / df.
5) The fifth column shows the F statistic, which is the ratio of the mean squares.
6) The sixth column shows the *p- value*, which is derived from the cumulative distribution function of F.

If the *p-value* is smaller than a certain limit, say 0.05, the null hypothesis is rejected. Otherwise, the null hypothesis is accepted with probability *p*.

The results appear in Table 4 confirms that the computational scales of all 36 APs in every group are of the same distribution.

Table 4. ANOVA table of 35 mix-relationship groups

Source	SS	df	MS	F	Pr > F
Rows	1252.05	34	36.825	1.08	0.3463
Error	11950934	349965	34.1489		
Total	11952186	349999			

To achieve more accurate results, we integrate all the 35 groups and fit the result with a statistical model. In this study, a log-logistic model is selected and the quantile plot is shown in Fig. 5.

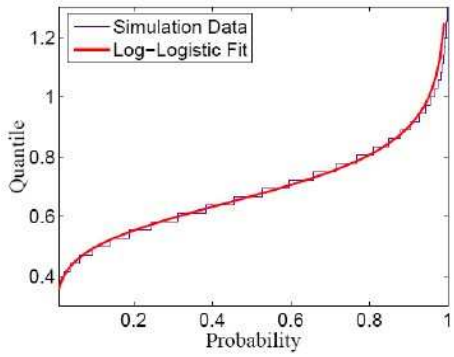

Figure 5. The Quantile Plot of the Pooled Data

It is observed that the result agrees nicely with the simulation work reported in [6]. The computation scale of mixed relationship is a log-logistic distribution rather than a normal distribution. This is because the means and variances of different relationship groups are different. The APs with small mean values tend to bias the probability density function curve to the left from a normal distribution.

Our goal is to find out an upper limit of the computation scale. It is natural that

$$CS_{\text{mixed relationship}} < CS_{\max\,(\text{single relationship})}$$

where *CS* stands for computational scale. Hence, the upper limit of the DHA is the upper limit of the 95% CI of the computation scale which is bounded by the integer ceiling of $(0.7438\times I+1.96\times 0.0326\sqrt{I})$, where *I* is the number of APs in the network.

3. CONCLUSION

In this paper, we analyze the computational scale of a typical RCA algorithm, namely DHA, and compare it with the simulation results. Then we demonstrate the effectiveness of the statistical method in predicting the computational scale of DHA. For a network of *I* APs, by extrapolating the simulation results based on statistical analysis, we find that the upper limit of the 95% CI of the computation scale of the network is bounded by integer ceiling of $(0.7438\times I+1.96\times 0.0326\sqrt{I})$. This estimate is within the region of $O(I)$ and is much smaller than the complete combination, namely $O(3^I)$.

APPENDIX

In this appendix, we present a simplification of the statistics model for the relationship matrix. The relationship matrix includes 50 rows, corresponding to 50 different relationships. However, in calculating the computational scale, some relationships yield similar results and therefore can be combined as one case. The MATLAB software tool, ANOVA, is used to analyze the statistical properties of the different relationships. Those relationships having similar properties are combined. The results are summarized in Table 5.

Table 5. ANOVA test result of combining similar relationship cases

Group No.	1	2	3	4	5	6	7	8	9
Combined cases	44–50	39–43	34–38	29–33	26–28	23–25	17–21	15,16	9,10
ANOVA test result (p-value)	0.807	0.788	0.955	0.055	0.455	0.921	0.892	0.441	0.456

No more cases can be combined. Consequently, we have the simplified 22 groups as shown in Table 6.

Table 6. Simplified relationship groups

	a vs b	b vs c	c vs d	a+b vs c	a+c vs d	b+c vs d	μ	σ		a vs b	b vs c	c vs d	a+b vs c	a+c vs d	b+c vs d	μ	σ
1	e	e	e	g			0.60	0.018	12				e			0.68	0.026
2	l	e	e	e			0.52	0.014	13				g			0.72	0.028
3				g			0.67	0.023	14	l	e	l	e			0.58	0.019
4	e	l	e	l	e		0.60	0.023	15				g			0.72	0.029
5				l	g		0.68	0.025	16	e	l	l	l		l	0.60	0.024
6				e			0.66	0.024	17				l		eg	0.68	0.025
7				g			0.69	0.026	18				e			0.67	0.025
8	e	e	l	e			0.39	0.015	19				g			0.72	0.029
9				g			0.67	0.023	20	l	l	l	l			0.71	0.028
10	l	l	e	l	e		0.66	0.026	21				e			0.70	0.028
11				l	g		0.70	0.027	22				g			0.75	0.033

REFERENCES

[1] K. Leung and B. Kim, 2003. Frequency assignment for multi-cell IEEE 802.11 wireless networks. *IEEE Vehicular Tech. Conf. (VTC)*, Orlando, USA.

[2] M. Yu, H. Luo, and K. Leung, July 2006. A dynamic radio resource management technique for multiple APs in WLANs. *IEEE Trans. on Wireless Comms.*, vol. 5, no. 7, pp. 1910–1920.

[3] Y. Zhao and K. K. Leung, 2006. Adaptive channel allocation for IEEE 802.11 wireless LANs. *The 12th European Wireless 2006 (EW2006) Conference*, Athens, Greece.

[4] M. Yu, B. W. Kwan, and X. Ma, 2007. A framework of radio channel allocation strategy for WLANs with multimedia traffic support. *ACM Int. Wireless Comms. and Mobile Computing Conf. (IWCMC)*, Hawaii, USA, pp. 553–558.

[5] R. Bruno, 2002. IEEE 802.11 optimal performances: RTS/CTS mechanism vs. basic access. *The 13th Int. Symp. On Personal Indoor and Mobile Radio Communications (PIMRC)* , Lisboa, Portugal, vol. 4, pp. 1747–1751.

[6] X. Ma, M. Yu. and B. Kwan, 2008. A Comparison Study of Two Heuristic MinMax Algorithms for WLAN Channel Allocations. Submitted to *IEEE CASE'08*.