Computing functions over wireless networks

P. R. Kumar

Dept. of Electrical and Computer Engineering, and
Coordinated Science Lab
University of Illinois, Urbana-Champaign

Email: prkumar@illinois.edu
Web: http://decision.csl.illinois.edu/~prkumar
How to process information in the network in sensor networks?

(Or how to do data fusion over a sensor network?)

Or how to compute a function of data over a sensor network?

Or how to perform in-network information processing in a sensor network?)
Outline

- Difference between sensor networks and data networks 4
- Model of problem: Protocol model, Non-information theoretic model 5
- Sample of results: Average vs. Max 9
- More details of results 12
- Some information theoretic results for sensor networks 27
- References 34
Sensor networks

- **Examples of Tasks**
  - Environment monitoring
    - Determine the Average temperature: \( \frac{x_1 + x_2 + \ldots + x_n}{n} \)
  - Alarm networks
    - Determine the Max temperature: \( \text{Max } x_i \)

- Sensor networks are not just data networks with sensor measurements replacing files
  - They are application specific
  - Nodes need not just relay packets
    - They can discard, combine, process packets
    - Combination of computing and communication

- More generally: Consider a symmetric function \( F(x_1, x_2, \ldots, x_n) \)
  - E.g., Average, Mode, Median, Percentile, Max
  - Determined by Histogram or “Types”

- How should information be processed *in the network* to compute such functions?
  - This can also be regarded as *network coding for sensor networks*
Model of problem: Protocol model, Non-information theoretic (Giridhar & K ‘05)
Model of problem

- Multi-hop model for wireless communication (Giridhar & K ’05)
  - Collocated network
    » All nodes within range of all
  - Multi-hop random network (Penrose 1997, Gupta & K ’98)
    » Critical common range for connectivity of random graph
      \[ r(n) = \sqrt{\frac{\log n + c_n}{n}} \]
    » where \( c_n \to +\infty \). (Take \( r(n) = \sqrt{\frac{2 \log n}{n}} \))
- At time \( t \), sensor \( i \) takes a measurement \( x_i(t) \in \{1, 2, \ldots, D\} \)
- Fusion node needs to calculate \( F(x_1, \ldots, x_n) \) exactly
- Non-information theoretic formulation
Model of problem

- Protocol Model for wireless communication
  - Receiver should be outside other transmitters’ interference footprints
  - Communicate at rate $W$ bits/sec (Take $W = 1$ bit/sec wlog)
- At time $t$, sensor $i$ takes a measurement $x_i(t) \in \{1, 2, \ldots, D\}$
  - No probability distribution on $x_i(t)$’s
- Fusion node needs to calculate $F(x_1, \ldots, x_n)$ exactly
- Non-information theoretic formulation

- Two types of networks
  - Collocated network
    - Large range so all nodes can hear each other
  - Random network
    - $n$ nodes randomly distributed
    - Need range at no less than
    \[
    r(n) \geq \sqrt{\frac{\log n}{n}}
    \]
    for network to be connected
Definition of Computational Rate $R_{\text{max}}(n)$

- Block coding allowed

  $N$ measurements of node 1: $x_1 \rightarrow x_1(1), x_1(2), \ldots, x_1(N)$

  $N$ measurements of node 2: $x_2 \rightarrow x_2(1), x_2(2), \ldots, x_2(N)$

  $N$ measurements of node $n$: $x_n \rightarrow x_n(1), x_n(2), \ldots, x_n(N)$

  $F(x_1(1), \ldots, x_n(1)) \ldots F(x_1(N), \ldots, x_n(N))$

- If all $N$ functions computed in time $T$
- Then Computational Rate $R = \frac{N}{T}$

- Best Rate over all Strategies $S$ and block lengths $N$: $R_{\text{max}}(n) = \sup_{S,N} \frac{N}{TS(N)}$

(Giridhar & K ‘05)
Sample of results: Average vs. Max
The *Average* versus *Max*

- **Theorem (Giridhar & K ‘05):** The rate at which the *Average* can be harvested is $\Theta\left(\frac{1}{\log n}\right)$
  - Strategy
    - Tessellate
    - Fuse locally
    - Compute along a rooted tree of cells

- **Theorem (Giridhar & K ‘05):** The rate at which the *Max* can be harvested is $\Theta\left(\frac{1}{\log \log n}\right)$
  - Strategy: Take advantage of Block Coding
    - First node announces times of max values: $(1\ 1\ 1\ 1\ 1)$
    - Second node announces times of additional max values: $(1\ 1\ )$
    - Third node announces of yet more max values: $(1\ )$
Summary: Order of difficulty of computations

(1/n)  
Collocated network:  
Average, Mode, Type  

Data downloading

(1/log n)  
Random Multi-hop network:  
Average, Mode, Type  

Collocated network:  
Max

(1/loglog n)  
Random Multi-hop network  
Max  

(Giridhar & K ‘05)
More details of results
(Giridhar & K ‘05)
Results: A classification of functions

- **Divisible functions**
  - Amenable to divide and conquer
  - \( R_{\text{max}}(n) = \Theta\left(\frac{1}{\log |\mathcal{R}(F_n)|}\right) \) if \( \deg(G_n) = O(\log |\mathcal{R}(F_n)|) \)

- **Symmetric functions**
  \( F_n(x) = F_n(\pi(x)) \) for every permutation \( \pi \)
  - Data centric paradigm: Identity of node is not important, only its value

- **Type-sensitive functions**
  - Hard to compute
  - \( R_{\text{max}}(n) = \Theta\left(\frac{1}{n}\right) \) in collocated case, and \( R_{\text{max}}(n) = \Theta\left(\frac{1}{\log n}\right) \) random case

- **Type-threshold functions**
  - \( R_{\text{max}}(n) = \Theta\left(\frac{1}{\log n}\right) \) collocated case, and \( R_{\text{max}} = \Theta\left(\frac{1}{\log \log n}\right) \) random case

(Giridhar & K ‘05)
Examples (Giridhar & K ’05)

- **Data download problem**: \( F_n(x_1, \ldots, x_n) = (x_1, \ldots, x_n) \):
  - In collocated or random networks: \( R_{\text{max}}(n) = \Theta\left(\frac{1}{n}\right) \)

- **Histogram of frequencies or “Type”**: \( F_n(x_1, \ldots, x_n) = (z_1, z_2, \ldots, z_D) \)
  - Collocated case: \( R_{\text{max}}(n) = \Theta\left(\frac{1}{n}\right) \)
  - Random networks: \( R_{\text{max}}(n) = \Theta\left(\frac{1}{\log n}\right) \)
  - Special case: Any symmetric function \( F(x_1, x_2, \ldots, x_n) \)

- **Mean, Mode, Median, Majority**:
  - Collocated case: \( R_{\text{max}}(n) = \Theta\left(\frac{1}{n}\right) \)
  - Random case: \( R_{\text{max}}(n) = O\left(\frac{1}{\log n}\right) \)

- **Max, Min, Range, Occurrence of a value**:
  - Collocated case: \( R_{\text{max}}(n) = \Theta\left(\frac{1}{\log n}\right) \)
  - Random case: \( R_{\text{max}} = \Theta\left(\frac{1}{\log \log n}\right) \)
Definition of Rate $R_{\text{max}}(n)$

- Block coding allowed

\[ N \text{ measurements of node } 1: \ x_1 \rightarrow x_1(1), x_1(2), \ldots, x_1(N) \]
\[ N \text{ measurements of node } 2: \ x_2 \rightarrow x_2(1), x_2(2), \ldots, x_2(N) \]
\[ N \text{ measurements of node } n: \ x_n \rightarrow x_n(1), x_n(2), \ldots, x_n(N) \]

\[ F(x_1(1), \ldots, x_n(1)) \ldots \ldots F(x_1(N), \ldots, x_n(N)) \]

Compute all $N$ functions

- If computed in time $T$
- Then Rate $R = \frac{N}{T}$

- Best Rate over all Strategies $S$ and block lengths $N$: $R_{\text{max}}(n) = \sup_{S,N} \frac{N}{T^S(N)}$

- Bound on $R_{\text{max}}$: $R_{\text{max}}(n) \leq \frac{W}{\log |\mathcal{R}(F_n)|}$

(Giridhar & K ‘05)
Divisible functions

- Divisible functions:
  - There exists $F_S(x_i; i \in S)$ for every subset $S \subseteq \{1, 2, \ldots, n\}$
  - With $|\mathcal{R}(F_S)| \leq |\mathcal{R}(F_n)|$

- $F_S(x_S) = g_S(F_{S_1}(x_{S_1}), F_{S_2}(x_{S_2}), \ldots, F_{S_m}(x_{S_m}))$ for partition $\{S_1, \ldots, S_m\}$ of $S$

- **Theorem:** $R_{\text{max}}(n) = \Theta\left(\frac{1}{\log |\mathcal{R}(F_n)|}\right)$ if $\deg(G_n) = O(\log |\mathcal{R}(F_n)|)$ (Giridhar & K ‘05)

- **Special cases**
  - **Data Downloading:** $\deg(G_n) \leq O(\log |\mathcal{R}(F_n)|) = O(\log D^n) = O(n)$
  - So $R_{\text{max}}(n) = \Theta\left(\frac{1}{n}\right)$

  - **Histogram:** $|\mathcal{R}(F_n)| = \left(\frac{n+D-1}{D}\right)$ and $\left(\frac{n}{D}\right)^D < \left(\frac{n+D-1}{D}\right) < (n+1)^D$

  - So $\deg(G_n) = \Theta(\log n) = \Theta(\log |\mathcal{R}(F_n)|)$ for Random networks
  - Hence $R_{\text{max}}(n) = \Theta\left(\frac{1}{\log n}\right)$ for Random networks (Giridhar & K ‘05)
Proof of $R_{\text{max}}(n) = \Theta\left(\frac{1}{\log |\mathcal{R}(F_n)|}\right)$ for Divisible Functions

- Tessellate into square cells of area $r^2/2$
- Neighboring occupied cells can communicate with each other
- Form a tree rooted at fusion center out of occupied connected cells
- Choose a relay node in each occupied cell and a parent in the next cell towards the root
- Locally compute and pass on along tree to root
  - Collect data from $\text{deg}(G_n)$ nodes within cell
  - Collect functional value of $\log|\mathcal{R}(F_n)|$ bits from bounded number of child cells
  - Pass on functional value of $\log|\mathcal{R}(F_n)|$ bits to parent cell
- All operations can be performed in $\Theta(\log |\mathcal{R}(F_n)|)$ time

- $R_{\text{max}}(n) = \Theta\left(\frac{1}{\log |\mathcal{R}(F_n)|}\right)$
- Constructive strategy
Symmetric functions

- Symmetric functions depend only on type $z = (z_1, z_2, \ldots, z_D)$
  - where $z_i =$ Number of occurrences of $i$ in $\{x_1, x_2, \ldots, x_n\}$
  - $F_n(x_1, x_2, \ldots, x_n) = \bar{F}_n(z_1, z_2, \ldots, z_D)$

- Type-sensitive functions
  - There is a $0 < c < 1$ such that a fraction $c$ of values $(x_1, x_2, \ldots, x_{cn})$ is never enough to pin down the value of the function $F_n$
  - Examples: Mean, Median, Mode, Majority

- Type-threshold functions
  - Only want to know whether each $z_i$ exceeds a threshold $z_i^*$
  - There is a threshold vector $z^* = (z_1^*, z_2^*, \ldots, z_D^*)$ such that $\bar{F}_n(z) = \bar{F}_n(z \land z^*)$ for all $n$
  - Examples: Max, Min, Range, Occurrence of value

(Giridhar & K ‘05)
Collision-free strategies in collocated case

- Every node knows when to transmit based on what it hears on channel
- The content of the packet it transmits depends on what it heard, as well as its own information

- Node $g_1$ transmits packet $P_1(x_{g_1})$
- Node $g_2(P_1(x_{g_1}))$ transmits packet $P_2(P_1(x_{g_1}), x_{g_2})$
- Node $g_3(P_1(x_{g_1}), P_2(P_1(x_{g_1}), x_{g_2}))$ transmits packet $P_3(P_1(x_{g_1}), P_2(x_{g_2}), x_{g_3})$
- ........

- Note: We are not allowing information transmission to occur through collisions

(Giridhar & K ‘05)
\[ R_{\text{max}}(n) = \Theta \left( \frac{1}{n} \right) \] for Type-sensitive functions in collocated case

- Wlog suppose \( D=2 \)
- Initially, \( x_{g1} \) is in the set \( S^0_{g1} \) with cardinality \( |S^0_{g1}| = 2^N \)
- After first transmission, \( x_{g1} \) can be in one of two sets depending on whether it transmits 0 or 1
  - Let the transmission correspond to the larger set, call it be \( S^1_{g1} \)
  - \( |S^1_{g1}| \geq 1/2 |S^0_{g1}| \)
- After \( t \)-th transmission of node \( k \), let \( x_k \) lie in \( S^t_k \) with \( |S^t_k| \geq 1/2 |S^{t-1}_k| \)
- So at the end, uncertainty set is: \( |S_1 \times S_2 \times \ldots \times S_n| \geq 2^{nN-T} \)

Thus at least \( nN-T \) places in the \( nN \) values \((x_1, x_2, \ldots, x_n)\) are undetermined
- However to compute \( F_n(x(1), x(2), \ldots, x(N)) \), at least \( cnN \) values are needed
- So \( nN-T \leq (1-c)nN \)
- So \( T \geq cnN \)
- Hence \( R = \frac{N}{T} \leq O \left( \frac{1}{cn} \right) \)

Thus \( R_{\text{max}}(n) = O \left( \frac{1}{n} \right) \) for collocated case (Giridhar & K ‘05)
Consider “Max” function (argument can be generalized)

Threshold vector = (1,1,...,1)

Lower Bound

- Take block length $N = \ell n > n$
- Node 1 transmits its locations of the $N_1$ 1’s in $\{x_1(1), x_1(2), \ldots, x_1(N)\}$
- Node 2 transmits the $N_2$ “new” 1’s in its list $\{x_2(1), x_2(2), \ldots, x_2(N)\}$
- Node 3 transmits the $N_3$ “new” 1’s in its list $\{x_3(1), x_3(2), \ldots, x_3(N)\}$

- To describe $N_i$ takes log $N$ bits
- To describe the locations of $N_i$ 1’s requires $\log \left( \frac{N - \sum_{j<i} N_j}{N_1} \right)$ bits
- So $T = n \log N + \sum_i \log \left( \frac{N - \sum_{j<i} N_j}{N_1} \right)$

Maximized when $N_i = N/n$. Use $\left( \frac{N}{n} \right)^n \leq \prod_i \left( \frac{n - i + 1}{\ell} \right)^n < (\ell/n)^{\ell n} = (ne)^{\ell n}$

- Thus: $R = \frac{N}{T} = \Omega \left( \frac{1}{\log n} \right)$

(Giridhar & K ‘05)
Upper bound $R_{\text{max}}(n) = O\left(\frac{1}{\log n}\right)$ in collocated case ....

- Take $N > 2n$. Consider

\[
x_1(1), x_1(2), \ldots, x_1(N) \quad \leftarrow \text{Exactly } N/2^n \text{ } 1\text{'s in } x_1 \\
x_2(1), x_2(2), \ldots, x_2(N) \quad \leftarrow \text{Exactly } N/2^n \text{ } 1\text{'s in } x_2 \\
x_n(1), x_n(2), \ldots, x_n(N) \quad \leftarrow \text{Exactly } N/2^n \text{ } 1\text{'s in } x_n
\]

At most one 1     .....At most one 1

- Claim: Each such $x$ produces a unique set of transmissions $P_1, P_2, \ldots, P_T$
  - Suppose not. Then there are two: $x$ and $y$ which produce same transmissions
  - They differ in some $x_k \neq y_k$
  - Then $\{x_1, x_2, \ldots, x_{k-1}, y_k, x_{k+1}, \ldots, x_n\}$ also produces same transmissions
    since node $k$ hears the same under $x_k$ or $y_k$ and so reacts the same
  - But this has different “Max” values from $x$
  - Thus “Max” functions are not determined from transmissions

(Giridhar & K ‘05) 22/37
Finishing the proof for the “Max” function

- Number of such vectors $x = \prod_{1 \leq i \leq n} \left( N - \frac{(i-1)N}{2n} \right) > \left( \frac{N}{2n} \right)^n > (n - 1)^N$

- So $2^T > (n - 1)^N$

- So $T > N \log(n-1)$

- So $R = \frac{N}{T} \leq \frac{1}{\log(n-1)}$

- Hence $R_{\text{max}}(n) = O\left( \frac{1}{\log n} \right)$

- This proves $R_{\text{max}}(n) = \Theta\left( \frac{1}{\log n} \right)$

(Giridhar & K ‘05)
Generalizing to any Type-threshold function for collocated case

- Feasibility of $R_{\text{max}} = \Omega\left(\frac{1}{\log n}\right)$
  - Node $i$ sends only list of 1’s for values which threshold has not been attained

- Upper bound of $R_{\text{max}}(n) = O\left(\frac{1}{\log n}\right)$
  - There exist a threshold vector such that
    $$\vec{F}_n(z_1 - 1, z_2, \ldots, z_D) \neq \vec{F}_n(z_1, z_2, \ldots, z_D)$$
    for all $n \geq \sum_i z_i$
  - Now consider an $x$ which has
    » $z_1$-1 vectors of the form $(1,1,\ldots,1)$
    » $z_2$ vectors of the form $(2,2,\ldots,2)$
    » $z_3$ vectors of the form $(2,2,\ldots,2)$
    » ...
  - Remaining $n \geq (\sum z_i) - 1$ have 1’s or 2’s only
  - Now problem is reduced to a “Max”

(Giridhar & K ‘05)
Random networks: Type sensitive networks $R_{\text{max}}(n) = \Theta\left(\frac{1}{\log n}\right)$

- Theorem (Giridhar & K ’05)
  - Type-sensitive functions: $R_{\text{max}}(n) = \Theta\left(\frac{1}{\log n}\right)$

- Proof
  - Tessellate unit area domain into squares of area $A = (\Delta r)^2/2$
  - Transmissions are local within square
  - Assume “Genie” communicates all messages instantaneously to all nodes
  - We know at least $cnN$ transmissions are needed
  - At least one square has greater than $cnNA$ receptions
  - However only one node can receive at a time in a square
  - So $T \geq cnNA = cnN(\Delta r)^2/2$
  - But $r \geq \sqrt{\frac{\log n}{n}}$ for connectivity
  - So $T \geq c’N \log n$
Type-threshold functions

- **Theorem (Giridhar & K ’05)**
  - Type-threshold functions: \( R_{\text{max}} = \Theta\left(\frac{1}{\log \log n}\right) \)

- **Proof**
  - Consider “Max” for simplicity
  - Tessellate unit area domain into squares of area \( A = (\Delta r)^2 / 2 \)
  - Some square has greater than \( nA \) nodes
  - Suppose all nodes outside this square have value 0
  - Then we need to compute “Max” of \( nA \) nodes
  - We need \( cN \log (nA) \) transmissions
  - Only one node can receive at any given time
  - So \( T \geq cN \log (nA) \)
  - But \( r \geq \sqrt{\frac{\log n}{n}} \) s needed for connectivity
  - So \( A \geq \frac{\log n}{n} \)
  - So \( T \geq N \Omega(\log \log n) \)
  - So \( R_{\text{max}} = O\left(\frac{1}{\log \log n}\right) \)
  - Achievability can be proved by using tree gathering
Some information theoretic results for sensor networks
Complexities of function computation over wireless networks

- Slepian-Wolf Theorem (‘73)
  - Total information fusion over wires from correlated sources

- Several other complexities in sensor networks

- Wireless nodes
  - There are no independent links: Sources share channel
  - Multiple access problem

- Also, sensors can communicate with each other and thus cooperate

- Source-channel separation does not hold

- Only a function needs to be computed, not all information

- Little is known in a pure information theoretic setting
Consider a network $p(y_2(t), y_3(t), ..., y_n(t) \mid x_1(t), x_2(t), ..., x_{n-1}(t))$

A feasible rate

$$R < \max_{p(x_1, ..., x_{n-1})} \min_{2 \leq k \leq n} I(X_1, \ldots, X_{k-1}; Y_k \mid X_k, \ldots, X_{n-1})$$

Generalization of Cover and El Gamal ‘79
Multiple access channel (Ahlswede ‘71, Liao ‘72)

- Consider the multiple access channel

\[ p(y_3(t)|x_1(t), x_2(t)) \]

- The capacity region

\[
\begin{align*}
R_1 &< I(X_1; Y_3|X_2) \\
R_2 &< I(X_2; Y_3|X_1) \\
R_1 + R_2 &< I(X_1, X_2; Y_3)
\end{align*}
\]

for any \( p(x_1)p(x_2) \)
A multiple source multiple relay network (Xie and K ‘07)

- Consider the network

\[ p(y_3(t), y_4(t), y_5(t) | x_1(t), x_2(t), x_3(t), x_4(t)) \]

- A feasible rate vector is

\[
\begin{align*}
R_1 &< I(X_1; Y_3 | X_3) \\
R_2 &< I(X_2; Y_4 | X_4)
\end{align*}
\]

and

\[
\begin{align*}
R_1 &< I(X_1, X_3; Y_5 | X_2, X_4) \\
R_2 &< I(X_2, X_4; Y_5 | X_1, X_3) \\
R_1 + R_2 &< I(X_1, X_3, X_2, X_4; Y_5)
\end{align*}
\]

for some joint distribution \( p(x_1, x_3)p(x_2, x_4) \).
A feasible rate for a sensor network with a sink

- Consider the network \( p \left( y_1(t), \ldots, y_n(t) \mid x_1(t), \ldots, x_n(t) \right) \)

- Feasible rate region for acyclic choice of routes

\[
R^{(1)} < I(U^{(1)}_{s_1}, U^{(1)}_{r_1}, Y_d|U^{(1)}_d, U^{(2)}_{s_2}, U^{(2)}_{r_2}, U^{(2)}_{r_3}, U^{(2)}_d, U^{(3)}_{s_3}, U^{(3)}_{r_3}, U^{(3)}_d) \\
R^{(2)} < I(U^{(2)}_{s_2}, U^{(2)}_{r_2}, U^{(2)}_{r_3}, Y_d|U^{(2)}_d, U^{(1)}_{s_1}, U^{(1)}_{r_1}, U^{(1)}_d, U^{(3)}_{s_3}, U^{(3)}_{r_3}, U^{(3)}_d) \\
R^{(3)} < I(U^{(3)}_{s_3}, U^{(3)}_{r_3}, Y_d|U^{(3)}_d, U^{(1)}_{s_1}, U^{(1)}_{r_1}, U^{(1)}_d, U^{(2)}_{s_2}, U^{(2)}_{r_2}, U^{(2)}_{r_3}, U^{(2)}_d)
\]

\[
R^{(1)} + R^{(2)} < I(U^{(1)}_{s_1}, U^{(1)}_{r_1}, U^{(2)}_{s_2}, U^{(2)}_{r_2}, U^{(2)}_{r_3}, Y_d|U^{(2)}_d, U^{(2)}_d, U^{(3)}_{s_3}, U^{(3)}_d, U^{(3)}_d, U^{(3)}_d) \\
R^{(1)} + R^{(3)} < I(U^{(1)}_{s_1}, U^{(1)}_{r_1}, U^{(3)}_{s_3}, U^{(3)}_{r_3}, Y_d|U^{(3)}_d, U^{(3)}_d, U^{(3)}_{s_3}, U^{(3)}_d, U^{(3)}_d, U^{(3)}_d) \\
R^{(2)} + R^{(3)} < I(U^{(2)}_{s_2}, U^{(2)}_{r_2}, U^{(2)}_{r_3}, U^{(3)}_{s_3}, U^{(3)}_{r_3}, Y_d|U^{(3)}_d, U^{(3)}_d, U^{(3)}_{s_3}, U^{(3)}_d, U^{(3)}_d, U^{(3)}_d)
\]

- Maximize over \( p (u^{(1)}_{s_1}, u^{(1)}_{r_1}, u^{(1)}_d)p (u^{(2)}_{s_2}, u^{(2)}_{r_2}, u^{(2)}_{r_3}, u^{(2)}_d)p (u^{(3)}_{s_3}, u^{(3)}_{r_3}, u^{(3)}_d) \)

- Based on combination of backward decoding for relay channel and multiple access channel

(Xie and K ’07) 32/37
Kramer, Gastpar and Gupta ‘03 have determined exact capacity region for relay channel with phase fading for some geometries.

**Theorem** (Xie and Kumar ‘07)

- Phase fading unknown to transmitter
- Node 5 far away from other nodes

The capacity region is:

For node 3,

\[
R^{(1)} < \log \left( 1 + \frac{P_1/d_{13}^\alpha}{N_3 + P_2/d_{23}^\alpha + P_4/d_{43}^\alpha} \right),
\]

(22)

for node 4,

\[
R^{(2)} < \log \left( 1 + \frac{P_2/d_{24}^\alpha}{N_4 + P_1/d_{14}^\alpha + P_3/d_{34}^\alpha} \right),
\]

(23)

and for node 5,

\[
\begin{aligned}
R^{(1)} &< \log \left( 1 + \frac{P_1/d_{15}^\alpha + P_3/d_{35}^\alpha}{N_5} \right), \\
R^{(2)} &< \log \left( 1 + \frac{P_2/d_{25}^\alpha}{N_5} \right), \\
R^{(1)} + R^{(2)} &< \log \left( 1 + \frac{P_1/d_{15}^\alpha + P_2/d_{25}^\alpha + P_3/d_{35}^\alpha + P_4/d_{45}^\alpha}{N_5} \right),
\end{aligned}
\]

(24, 25, 26)
References-1


References-3


Attribution-Noncommercial-No Derivative Works 3.0 Unported

You are free:

- to Share — to copy, distribute and transmit the work.

Under the following conditions:

- Attribution — You must attribute this work to P. R. Kumar (with link).
  http://decision.csl.illinois.edu/~prkumar/html_files/talks.html

- Noncommercial — You may not use this work for commercial purposes.

- No Derivative Works — You may not alter, transform, or build upon this work.

With the understanding that:

- Waiver — Any of the above conditions can be waived if you get permission from the copyright holder.

- Other Rights — In no way are any of the following rights affected by the license:
  - Your fair dealing or fair use rights;
  - The author's moral rights;
  - Rights other persons may have either in the work itself or in how the work is used, such as publicity or privacy rights.

Notice — For any reuse or distribution, you must make clear to others the license terms of this work. The best way to do this is with a link to this web page.

This is a human-readable summary of the Legal Code (the full license).