

## TCP-Cherry+: 継続的なエラーが発生する 衛星 IP ネットワークのための TCP-Cherry の改良

内海哲史<sup>†1</sup>

富樫洸<sup>†1</sup>

S.M. Salim Zabir<sup>†2</sup>

有線ネットワークと異なる特性をもつ衛星 IP ネットワークにおける TCP の改良技術として, TCP-Cherry が提案されている. TCP-Cherry ではリンクエラー率の高い衛星ネットワークの輻輳制御のアルゴリズムとして Fast-Forward Start, First-Aid Recovery を持つ. これらのアルゴリズムではサブプリメントセグメントと呼ばれる低優先度セグメントを用いた. これによってデータを送りながら, 利用可能なリンク容量を素早く把握することができ, 従来の TCP 輻輳制御に比べウィンドウサイズを素早く最適なサイズに調整できることを可能とした. しかし, シャドローイングが発生する都市環境下で TCP-Cherry を用いてモバイル衛星通信を行った場合は, なお性能低下が起こる. そこで本論文では, シャドローイング発生する環境下においても活用できる新たな通信方式 TCP-Cherry+ の提案を行う. また, 本提案方式による性能向上を示すために, TCP-Cherry と TCP-Cherry+ をシミュレータ上に実装し評価する.

### TCP-Cherry+: Improvement of TCP-Cherry for Satellite IP Networks with Continuous Errors

SATOSHI UTSUMI,<sup>†1</sup> KO TOGASHI<sup>†1</sup>  
and S.M. SALIM ZABIR<sup>†2</sup>

TCP-Cherry, an enhancement of TCP that we have devised recently, aims at improving TCP performance over satellite IP networks. In TCP-Cherry, two new algorithms, Fast-Forward Start and First-Aid Recovery, have been proposed for congestion control in satellite network with long propagation delays and high link errors. Our algorithm use low-priority segments, we named supplement segments, which prove the available bandwidth in the network for the TCP connections along with carrying new data blocks. In this paper, we propose new congestion control scheme for TCP-Cherry in mobile satellite networks with persistent errors due to shadowing and shadowing. We name the scheme as TCP-Cherry+. Simulation results show that TCP-Cherry+ yields up to a maximum improvement of 25% with TCP-Cherry in shadowing scenario.

### 1. はじめに

モバイル衛星ネットワークは有線ネットワークと異なり, 地上に有線設備や基地局を必要としないため, 離島や山間部といった有線設備の構築が困難である地域におけるデジタルディバイドの解消が可能である. また, 基地局が災害により停止した場合にも通信が可能となるため, 衛星による通信が期待されている. しかし, モバイル衛星ネットワークは, 無線通信のために大気, 電離層及び天候の状態による影響を受けるほか, 都市環境下ではシャドローイングやフェージングといった外的要因を受ける. しかしながら, 従来の TCP は, 地上ネットワークでの通信を想定したプロトコルであるため, 衛星 IP ネットワークのような高遅延無線環境においては, ウィンドウサイズの増加に要する時間が長くなることから, 十分なスループットが得られないと指摘されている. これまで, 著者らは衛星通信の特性を考慮した輻輳制御方式として, TCP-Cherry を提案し, コネクション確立から素早くウィンドウサイズを増加させることを可能とした. しかし, TCP-Cherry は, シャドローイングが発生する都市環境下では性能がなお低下する. 本論文ではその問題点を改善した, TCP-Cherry+及び同プロトコル中のアルゴリズム New Fast-Forward Start を提案する.

### 2. 関連研究

長い伝搬遅延時間を持つ衛星ネットワークにおける TCP 輻輳制御の方式として, TCP-Cherry<sup>1)</sup> が提案されている. TCP-Cherry ではサブプリメントセグメントという優先度の低いセグメントを活用し輻輳制御を行う. サプリメントセグメントは Fast-Forward Start, First-Aid Recovery の二つのアルゴリズム中に用いられる. 本節では, サプリメントセグメントと Fast-Forward Start について概説する.

#### 2.1 サプリメントセグメント

TCP-Peach<sup>2)</sup>, TCP-Peach+<sup>3)</sup> では, 低優先度の dummy segments, NIL segment を送信し, それらの確認応答より利用可能なリンク容量を把握して輻輳制御を行う. しかしながら, dummy segment, NIL segment に使用されるセグメントは, 既に送信したデータセグ

<sup>†1</sup> 鶴岡工業高等専門学校

Tsuruoka National College of Technology

<sup>†2</sup> フランステレコム株式会社

France Telecom Japan

```

Select_Data4Supple()
  if just before Fast-Forward Start / First-Aid Recovery
    Calculate_Skip_Window( );
    SkipData = Get_Next_Send(ThreshData,  $\sigma+1$ );
    seqno4supple = SkipData;
  else
    seqno4supple = Get_Next_Send(seqno4supple, 1);
  end;
end;

```

図 1 The Data Selection Algorithm for TCP-Cherry.

```

Calculate_Skip_Window()
  if just before Fast-Forward Start
     $\sigma = \text{maxcwnd}$ ;
  end;
  if just before First-Aid Recovery
     $\sigma = \text{cwnd} * (\text{MaxRTT} / \text{RTT})$ ;
  end;
end;

```

図 2 The Skip Size Determination Algorithm for TCP-Cherry.

メントから選ばれるため、dummy segments, NIL segment の受信は重複セグメントの受信であり、オーバーヘッドとなる。この問題に対して TCP-Cherry では、Select\_Data4Supple (図 1), Calculate\_Skip\_Window (図 2) のアルゴリズムを用いて低優先度のセグメントを未送信のセグメントから選択することで、スループットを向上させている。データセグメントと未送信のセグメントの差を Gap Segment という。この選択された低優先度のセグメントがサプリメントセグメントである。

## 2.2 Fast-Forward Start

Fast-Forward Start は、TCP Reno における Slow Start の位置づけである。Slow Start では、長い伝搬遅延特性を持つ衛星ネットワークでは、ウインドウサイズの増加が遅いという問題があったが、Fast-Forward Start のアルゴリズム (図 3) では素早くリンク容量を把握し最適なウインドウサイズを求め、素早く Congestion Avoidance の段階に移行することが可能となっている。

```

Fast-Forward_Start()
   $\text{cwnd} = 1$ ;  $\tau = \text{RTT} / \text{maxcwnd}$ ;
  send(Data_Segment);
  for  $i = 1$  to  $(\text{maxcwnd} - 1)$ 
    wait( $\tau$ ); send(Supplement_Segment);
  end;
end;

```

図 3 Fast-Forward Start Algorithm for TCP-Cherry.

## 3. 提案方式:TCP-Cherry+

本節では、初めに TCP-Cherry のタイムアウト後の動作を説明する。次に TCP-Cherry+ の動作並びに新しいアルゴリズムである NewCalculateSkipWindow, NewSelectData4Supple を説明する。

### 3.1 シャドーイング時の Fast-Forward Start

Congestion Avoidance の段階で、送信したデータセグメントに対する確認応答が到達せず、タイムアウトが発生した場合、Fast-Forward Start の段階に移行すると仮定する。図 4 は cwnd を 40 として通信を行っている際に、シャドーイングが発生した場合の様子である。maxcwnd は 50 としている。タイムアウトにより再送を行ったセグメント (図 4 では 1004(2)) に対しても確認応答が返信されなかった場合、指数バックオフによりタイムアウト値を二倍にして再度送信する (図 4 では 1004(3))。このような動作によって、図 4 のシャドーイング終了後の  $\alpha$  秒は通信可能であるにも関わらず、データが送信されない時間となっていた。また、Fast-Forward Start 後の Congestion Avoidance の段階では重複したセグメントを送信していることが確認された。図中では 1066~1102 セグメントが重複セグメントである。これは Gap Segment 数が小さかったことに起因すると考えられる。TCP-Cherry では上記の理由により、タイムアウト二回目以降では、Fast-Forward Start をやめ、Slow Start を開始する仕様となっている。

### 3.2 New Fast-Forward Start, New Select Data4Supple, New Calculate Skip Window

TCP-Cherry+ では、New Fast-Forward Start を用い、3.1 節で述べた空白の  $\alpha$  秒間での通信を可能にし、さらにセグメント重複の解決が図られている。図 4 と同様の条件下で、

TCP-Cherry+が通信を行う際の様子を図 5 に示す。なお、図中のシャドーイング前の通信は TCP-Cherry と同じなので割愛する。

New Select Data4Supple のアルゴリズムを図 6 に示す。このアルゴリズムではタイムアウト後に確認応答が到達するまで繰り返しサブリンクセグメントを選択するように改良されている。

New Calculate Skip Window のアルゴリズムを図 7 に示す。このアルゴリズムでは Gap Segment 数を計算している。先に述べたセグメント重複を防ぐため、Gap Segment を maxcwnd の二倍に設定している。

New Fast-Forward Start のアルゴリズムを図 8 に示す。New Fast-Forward Start では、ACK が返ってくるまで、サブリンクセグメントを繰り返し送信する。

TCP-Cherry+のフローチャートを図 9 に示す。

## 4. 評価

コンピュータシミュレーションによって、本提案方式を TCP-Cherry と比較しスループットが向上することを示す。

### 4.1 シミュレーションシナリオ

図 10 のようなモバイル GEO 衛星 IP ネットワークのシナリオにて、評価を行った。タイムアウト後の性能を計測するため、シャドーイングが通信中に 5 秒間発生する環境を設定した (図 11)。図 11 のシャドーイングの間隔  $X_s$  (秒) はセグメント損失率を 0 % から 50 % で変化させるために、 $X_s$  (秒) を調整する。例えば、セグメント損失率が 50 % のシミュレーションを行う場合は、 $X_s$  (秒) = 5s (秒) となる。ここで、シャドーイング中の通信はセグメント損失率 100 % であり、また、シャドーイング外の通信中ではセグメント損失率は 0 % であるとする。シミュレーションにはネットワークシミュレータ ns-2 を使用し、モバイル間のスループット (パケット/秒)、オーバーヘッド (%) 計算した。

- 他の設定詳細

フロー数 :  $N=1$

TCP セグメントサイズ : 1000 バイト

衛星リンクの容量 : 1300 セグメント/秒 (約 10Mbps)

衛星アップリンクのバッファサイズ :  $K=50$  セグメント

maxcwnd : 512 セグメント

受信側の TCP のバッファサイズ (rwnd) : 65536 セグメント

往復遅延時間 : RTT=550ms

### 4.2 性能評価

提案手法の評価を行い、TCP-Cherry とのスループット、オーバーヘッドによる性能の比較を行う。ここで TCP-Cherry+ の妥当性を証明する。

#### 4.2.1 スループットの評価

シミュレーション結果より得られたスループットを図 12 に示す。横軸をセグメント損失率、縦軸をスループットとしている。結果より、セグメント損失率が高い通信時では TCP-Cherry+ の方スループットが高く保たれていることが確認された。

#### 4.2.2 オーバヘッドの評価

図 13 にオーバーヘッドの計算結果を示す。横軸をセグメント損失率、縦軸をオーバーヘッドとしている。図 13 では TCP-Cherry+ の方が、オーバーヘッドが高くでている。これはシミュレーションのシャドーイングの時間を 5 秒と設定したため、タイムアウトが複数回発生し、TCP-Cherry では再送処理がスロースタートから行われたためと考えられる。TCP-Cherry+ のオーバーヘッドはセグメント損失率に対して単調増加であるが、セグメント損失率 50 % の通信においても僅か 0.2 % のオーバーヘッドであることがシミュレーションより確認された。これは、TCP-Cherry+ ではタイムアウトが複数回発生した後でも、少ないオーバーヘッドで素早く通信を回復することを可能にしていることを示している。

## 5. 結論

モバイル衛星通信は優れた可用性を備えており、今後有線ネットワークと併用しながら日常で使用されていくと考えられる。しかし、有線ネットワークでの使用を想定した TCP を衛星ネットワークに用いた場合満足な結果は得られていない。本論文では、TCP-Cherry のタイムアウト後の動作に改善を行い TCP-Cherry+ の提案を行った。また、提案手法の評価として、タイムアウトが発生する都市環境を想定し、ns-2 によるシミュレーションを行った。シミュレーションの結果スループット、オーバーヘッドからの性能評価において、TCP-Cherry+ は TCP-Cherry より優れた結果を示した。

## 参考文献

- 1) S. Utsumi, S. M. S. Zabir, N. Shiratori, "TCP-Cherry: A New Approach for TCP Congestion Control over Satellite IP Networks," Computer Communications, Elsevier, vol.31, June 2008, pp. 2541-2561.

- 2) Ian F. Akyildiz, Giacomo Morabito, and Sargio Palazzo “TCP-Peach:A New Congestion Control Scheme for Satellite IP Networks” IEEE/ACM transactions on networking, vol.9,no.3, June 2001
- 3) Ian F. Akyildiz, Xin Zhang, and Jian Fang “TCP-Peach+: Enhancement of TCP-Peach for Satellite IP Networks” IEEE Communications Letters, vol.6,no.7, July 2002

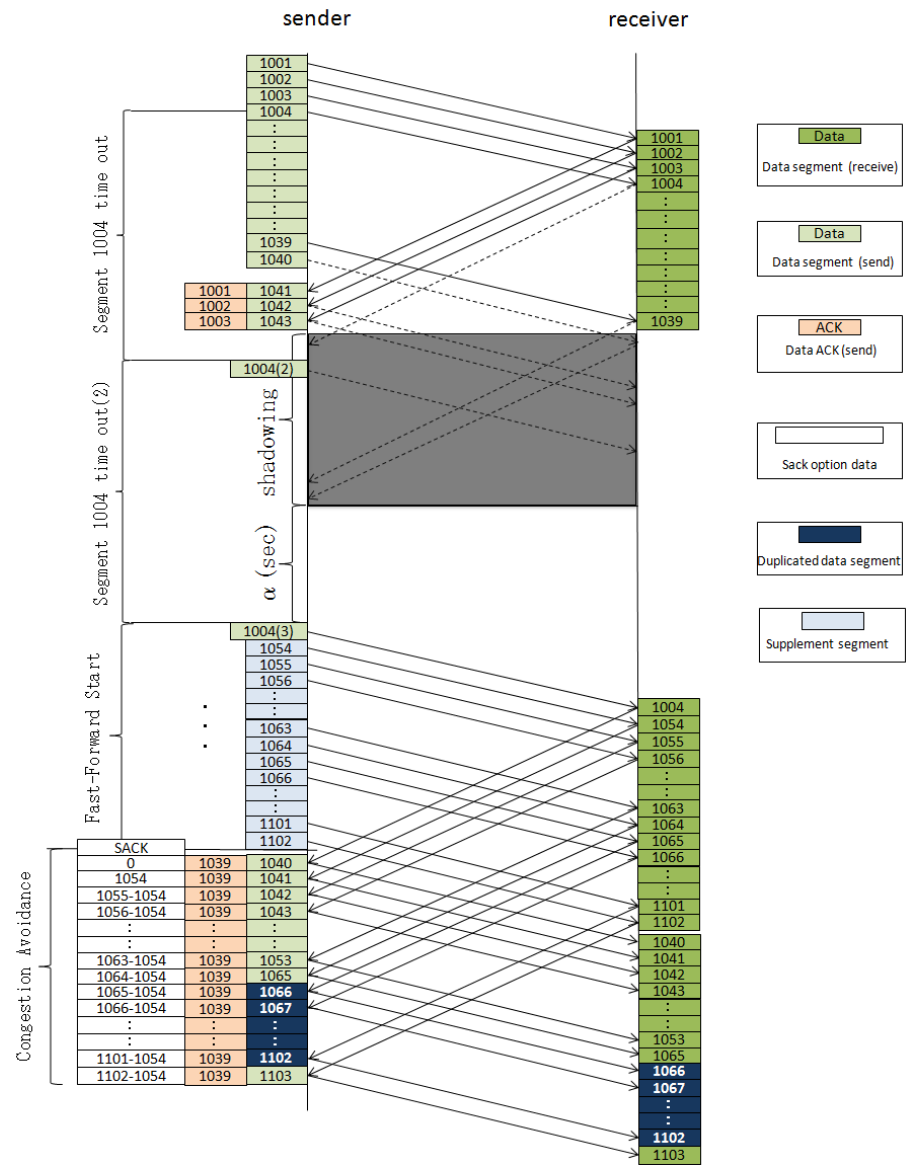


图 4 Behavior of Fast-Forward Start with Shadowing.

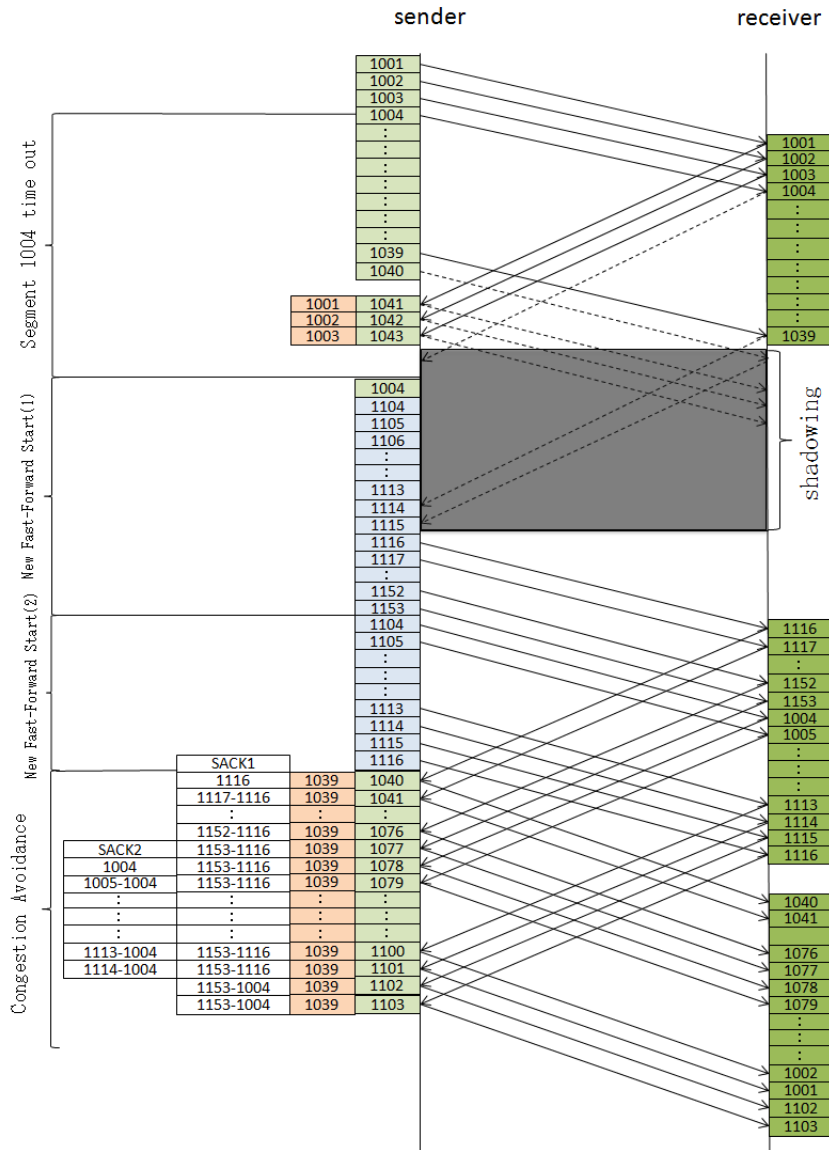
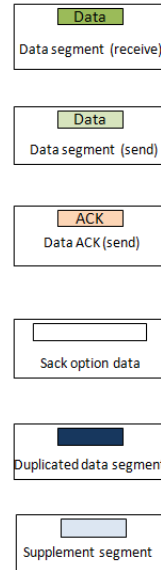


Figure 5 Behavior of New Fast-Forward Start with Shadowing.



### NewSelect\_Data4Supple()

```

if just before Fast-Forward Start / New Fast-Forward Start / First-Aid Recovery
    i = 0;
    NewCalculate_Skip_Window();
    SkipData = Get_Next_Send(ThreshData,  $\sigma+1$ );
    seqno4supple = SkipData;
else if New Fast-Forward Start
    i++;
    if (i > maxcwnd)
        i = 0;
        seqno4supple = SkipData;
    else
        seqno4supple = Get_Next_Send(seqno4supple, 1);
    end;
else
    seqno4supple = Get_Next_Send(seqno4supple, 1);
end;
end;
end;

```

Figure 6 The New Data Selection Algorithm for TCP-Cherry+.

### NewCalculate\_Skip\_Window()

```

if just before Fast-Forward Start
     $\sigma$  = maxcwnd;
end;
if just before New Fast-Forward Start
     $\sigma$  = cwndmax * 2;
end;
if just before First-Aid Recovery
     $\sigma$  = cwnd * (MaxRTT/RTT);
end;
end;
end;

```

Figure 7 The New Skip Size Determination Algorithm for TCP-Cherry+.

**NewFast-Forward\_Start()**

```

cwnd=1; τ=RTT/maxcwnd;
END=0;
send(Data_Segment);
while (END=0)
  wait(τ); send(Supplement_Segment);
  if (ACK_ARRIVAL)
    END=1;
  end;
end;
end;

```

图 8 New Fast-Forward Start Algorithm for TCP-Cherry.

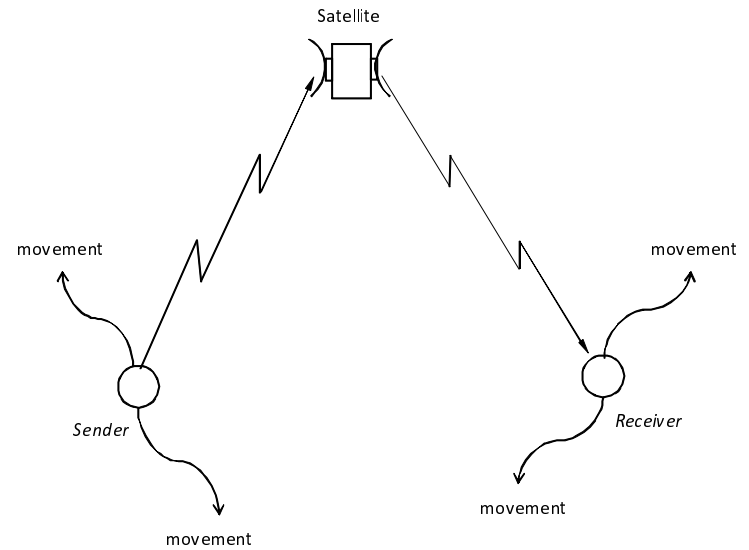


图 10 Satellite with Mobile Node Scenario.

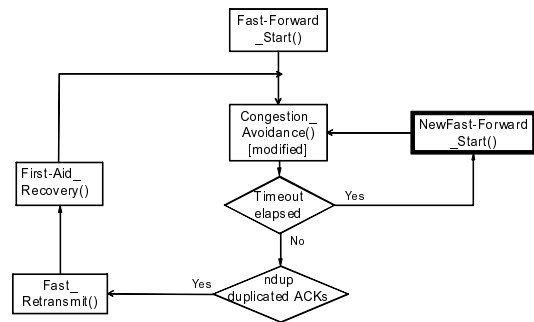


图 9 TCP-Cherry+ Scheme.

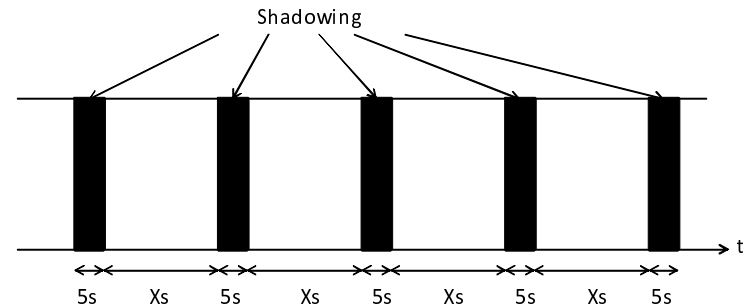


图 11 Shadowing Scenario.

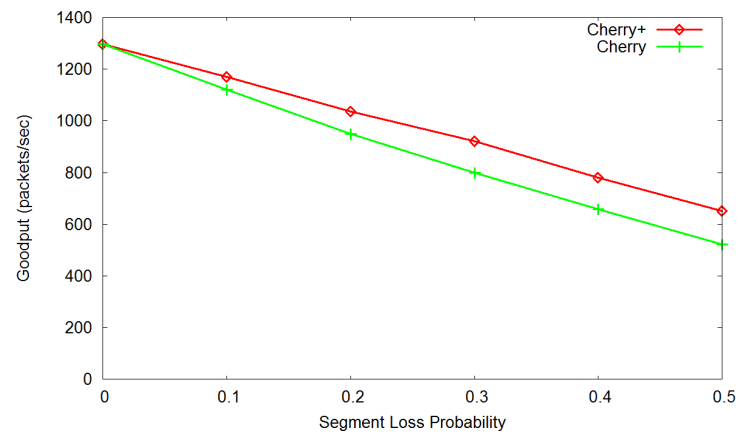


图 12 Goodput Comparison in the GEO Satellite(10Mbps) Network with Shadowing.

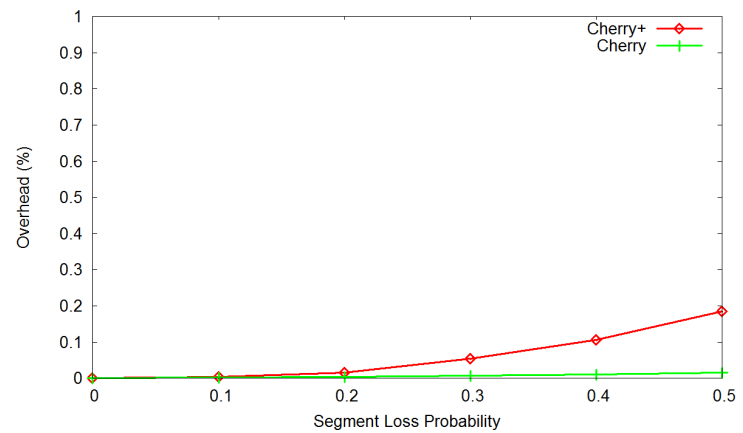


图 13 Overhead Comparison in the GEO Satellite(10Mbps) Network with Shadowing.