

NS-3 Simülasyon:

NS-3, ağ simülasyonu için akademik araştırma ve öğretim ihtiyaçlarını karşılamak üzere tasarlanmış ayrı, olay odaklı bir ağ simülatörüdür. NS-3 projesi, 2006 yılında başlayan ve NS3 yazılımının geliştirilmesinden sorumlu olan açık kaynaklı bir projedir. NS-3, NS-2'nin bir uzantısı değil, tamamen yeni bir simülatördür.

WAF

Python tabanlı olarak geliştirilmiş bir derleme aracıdır ve NS-3 sisteminin kendisi ve yazdığımız simülasyon kodu waf tarafından derlenir ve çalıştırılır.

scratch

Bu dizin genellikle kullanıcı komut dosyalarını depolar ve çalıştırılacak örnekleri NS-3'ün varsayılan komut dosyası depolama dizini olan bu dizine de kopyalayabilirsiniz ve komut dosyalarını derlemek ve çalıştırmak için WAF kullanırken, komut dosyasının diğer dizinlerdeki dosya adına eklenmesi gerekiyorsa, dizin eklemekten bir dizin scratch ekleyebilirsiniz

örnekler

Enerji, yönlendirme, kablosuz ağ vb. gibi birçok modülün kullanımı da dahil olmak üzere NS-3 tarafından sağlanan NS-3'ün nasıl kullanılacağına dair örneklerdir ve NS-3 dizinindeki örneklerle başlamak önemlidir. Bunlar arasında, öğretici dizinindeki örnekler yeni başlayanlar için çok uygundur.

DOC

Yardım belgeleri

Derleme

NS-3 derleme sonuçları, dosyaları derlemek için kullanılan paylaşılan kitaplıkları ve başlık dosyalarını içeren bir dizinde saklanır

SRC

Modülde bulunan kaynak kodunu ve diğer modüllerin kullanımını kaydetmek için kullanılan wscript dosya yapısı sabittir; Modül kodunun .cc ve .h dosyaları model dizininde bulunur, yardımcı dizin modülün karşılık gelen yardımcı sınıf kodunun kaynak dosyalarını depolamak için kullanılır, test dizini modül tasarımcısı tarafından yazılan modül test kodunu içerir, örnekler dizini modülü uygulayan örnek kodu içerir, doc dizini yardım belgesidir ve bindings dizini modül tarafından Python dilini bağlamak için kullanılır.

Aqua-Sim-NG:

Aqua-Sim NG çekirdeği, Aqua-Sim 1.0'da görülen yarı çift yönlü, uzun yayılma gecikmeleri ve akustik sinyal zayıflaması gibi temel su altı işlevlerini desteklemeye devam ediyor. Mevcut protokol desteği şunları içerir: (MAC katmanı:) BroadcastMAC, Sualtı ALOHA, CopeMAC, Sualtı FAMA, Sualtı Oluklu FAMA, GOAL, UWAN, RMac, Tmac, (Yönlendirme Katmanı:) DBR, Artık-DBR, Statik, Dinamik, Kukla, Taşma, DDoS-Kısıtlama, VBF ve VBVA.

NS3 Mimarisi Desteği

Temel Aqua-Sim yenilemesi ve API entegrasyonu

Örnek ve yardımcı komut dosyaları ekleme

Protokol Bağlantı Noktası

Kapsamlı protokol bağlantı noktası

Gerçek Dünya Özellikleri

Senkronizasyon desteği

Yerelleştirme desteği

Meşgul terminal modeli tümleştirilmesi

Dalgalı SNR

Gelişmiş kanal modeli desteği

İletim aralığı belirsizlik modülü

Güvenlik özellikleri ve modül desteği

Kanal katmanı için iz odaklı destek

Bilgi Merkezli Entegrasyon

Uyarlanmış Adlandırılmış Veri Ağı bileşenleri

Yardımcı ve test komut dosyaları entegre edildi

Özel protokollerin eklenmesi

Gelişmiş Kanal Modelleri

Çok kanallı simülasyonu, farklılaştırılmış gürültü üretici sınıflarını ve ek yayılma modeli desteğini desteklemek için genişletilmiş kanal modelleri. Buradaki amacımız, simülasyon sonuçlarımızda gerçek dünya senaryolarına daha iyi doğruluk göstermektir.

İz GÜdümlü Simülasyon

İzleme güdümlü bir simülasyon uygulayarak, test yinelemeleri boyunca belirli koşulları kontrol edebiliyoruz. Bunun bir örneği, bir okyanus testinden kanal okumalarını doğrudan simülatöre uygulamak ve her test boyunca verilen parametreleri göstermek olabilir. Bu, protokol testi sırasında kanal koşullarının aynı kalmasını sağlamak gibi durumlarda faydalı olabilir. Bu kanal tutarlılığı gerçek dünya senaryolarında garanti edilmese de, simülasyonları çalıştırırken ek kontrol ekler.

Senkronizasyon ve Yerelleştirme

Senkronizasyon ve lokalizasyon, doğru su altı simülasyonu sağlamak için iki kritik bileşendir. Geçmişteki Aqua-Sim çalışması, bu iki özellik için özel bir modülü desteklemiyor ve bu da yükü geliştiriciye yüklüyor. Bu çalışma için, menzil tabanlı veya menzilsiz yerelleştirme

desteđi gibi konum özelliklerini dahil ediyoruz. Zaman senkronizasyonu için, saat eğriliđini ve ofsetini daha iyi entegre etmek için özellikler içeriyoruz. Bu özelliklerin amacı, hem kullanıcılar hem de geliştiriciler için daha kolay protokol entegrasyonuna izin vermektir.

Gerçek Sistem Özellikleri

Gerçek sistem özellikleri, UWSN'lerde görülen senaryoları daha iyi taklit edecek kapsayıcı bir kategoridir. Kanal koşullarının belirsizliđi nedeniyle, bağlantı asimetrisi ve heterojenliđi uyguluyoruz. Bu, kanalın dalgalanan sinyal-gürültü oranları gibi koşulları simüle etmesine olanak tanır. Ayrıca, UWSN'lerdeki bu çevresel faktörler, daha dođru sonuçlar için dikkate alınması gereken akustik iletim aralıđı belirsizliđine yol açabilir. UWSN'lerde gördüğümüz bir diđer benzersiz özellik de meşgul terminal sorunudur. Bu yoğun terminal sorununa katkıda bulunan iki ana faktör, akustik iletişim hızı ve UWSN'lerin yarı çift yönlü doğasıdır. Bu bileşenleri Aqua-Sim Next Generation'da göstererek gerçek bir sistem simülasyonumuzu daha da güçlendirebiliriz.

Güvenlik Özellikleri

Diđer tüm ağ türleri gibi, UWSN de saldırılara karşı hassastır. Sıkışma, hizmet reddi, paket sahtekarlıđı, düden ve solucan deliđi bileşenlerini uygulayarak, potansiyel su altı güvenlik tehditlerini simüle etmeye yardımcı olabiliriz. Amacımız bu tip saldırıların kötü niyetli olarak oluşturulmuş node'lar tarafından daha kolay entegrasyonuna ve daha kolay analiz desteđine olanak sağlamak.

Bilgi Merkezli Entegrasyon

UWSN'ler için özel NDN protokollerinin yanı sıra uyarlanmış Adlandırılmış Veri Ađı (NDN) tekniklerini uyguluyoruz. Bu özellikleri uygulayarak, NDN'nin bu ortamda sahip olabileceđi iyileştirmeleri daha iyi tasvir edebilir ve değerlendirebiliriz. Bu entegrasyon, su altı için özelleştirilmiş genel NDN dahil edilmesi için ilgi alanı ve veri yapılandırılmış paketleri, Bekleyen Faiz Tabloları, Yönlendirme Bilgi Tabanları ve Bağlam Depolarının tanıtılmasından oluşur. Ek olarak, bu modülü fiziksel katmandaki bir paket demux aracılığıyla uyguluyoruz ve protokollerin simülatörümüzün işleyişini tamamen yenilemeden belirli NDN özelliklerini uygulamasına izin veriyoruz.

Aloha

ALOHA yöntemi, Hawaii Üniversitesi'nde, birden fazla kullanıcının radyo iletimi yoluyla ortak bir kanal üzerinden merkezi bir istasyona, büyük ölçüde koordinasyonsuz bir şekilde kısa veri paketleri göndermesi için bir araç olarak ortaya çıkmıştır . Merkez istasyon tüm alımlarını farklı bir kanalda yeniden yayınlıyacaktı, böylece gönderen kullanıcılar diđerleriyle birlikte kendi yayınlarını geri alacaktı. Kullanıcı, veri paketinin bir başkasıyla çarpıştığını gözlemlerse, bir süre sonra yeniden iletilir. Gönderen kendi paketini değiştirilmemiş olarak görürse, bu bir onay görevi görür.

ALOHA'nın iki ana kategorisi yarıksız ALOHA ve yarıklı ALOHA 'dur. Yuvasız ALOHA'da, iletilen bir paket herhangi bir başlangıç zamanında gönderilebilir ve çeşitli sürelerde olabilir. Oluklu ALOHA'da, sabit boyutlu senkronize zaman dilimleri vardır. Bir gönderen herhangi bir zaman diliminde gönderebilir ve veri paketlerinin tümü bir

zaman dilimi süresinden biraz daha küçük olmalıdır. Çarpışmalar, yarıksız ALOHA'da tam veya kısmi olabilir, ancak yarıklı ALOHA'da eksiksizdir. Kısmi bir çarpışma bile genellikle paketin yeniden iletilmesini gerektirir.

Yayın yöntemi, gönderen kullanıcının başarılı bir iletimden emin olması için yeterli bilgi sağlamıyorsa, başka bir seçenek de hedefin veya merkezi istasyonun bireysel bir onay sinyalini geri göndermesidir.

ALOHA hafif trafik durumlarında çok etkilidir; Gönderici tüm kanalı kullanabilir ve genellikle çarpışma olmadan çok hızlı bir şekilde gönderebilir. Daha yoğun trafiğe sahip olan ALOHA, verimlilik ve istikrar sorunlarından muzdariptir. Bu sorunları hafifletmek için çeşitli protokoller ve sinyal işleme yöntemleri icat edilmiştir. Bunlar arasında çarpışma çözünürlük algoritmaları, varış hızının ve yeniden iletim süresinin kontrolü, rezervasyon ALOHA, çok kanallı ALOHA, yakalama ALOHA ve çeşitlilik alımı yer alır. Ayrıca, çarpışma algılamalı taşıyıcı algılamalı çoklu erişim (CSMA/CD) kullanılarak yuvasız ALOHA'nın modifikasyonu, popüler Ethernet yerel alan ağ protokolünün temelini oluşturur.

İdealize Edilmiş Oluklu ALOHA İletişiminin Kantitatif Analizi

Oluklu ALOHA'nın kullanıldığını ve tüm kullanıcıların tüm iletimlerin sonuçlarını gözlemlediğini varsayalım. Modeller, (1) sonlu bir toplam mesaj varış hızına sahip sonsuz sayıda kullanıcı veya (2) sınırlı sayıda kullanıcı olduğunu varsaymıştır. Yeniden iletim için zaman gecikmesinin, her yeniden iletim için bağımsız olarak rastgele seçildiği varsayılır; İki çarpıştırıcı için aynı olsaydı, kesin olarak tekrar çarpışırlardı.

ALOHA'nın Verimliliğini Artırma

Verimliliğini artırmak için ALOHA iletişimini değiştirmenin üç yolu vardır:

1. Çarpışma sıklığını ortadan kaldırmak veya azaltmak için protokolleri kullanın.
2. Sinyal tasarımı, sinyal işleme ve/veya çeşitlilik alımı yoluyla çarpışmayı tolere edin.
3. İkili olmayan iletimi kullanarak yuva başına bit sayısını artırın.

İlk caddedeki teknikler şunları içerir: (a) yuvasız—yalnızca kanalda herhangi bir etkinlik algılanmadığında gönder, taşıyıcı algılamalı çoklu erişim (CSMA) olarak adlandırılır; (b) yuvasız—çarpışma algılama (CD) adı verilen bir çarpışma algılanırsa göndermeyi durdurun; (c) Yarıklı—çarpışma çözünürlüğü algoritmaları.

Avenue 1a ve 1b, yerel alan ağları için Ethernet protokolünde CSMA/CD olarak kullanılır. Kanalin algılanması, potansiyel göndericiler arasındaki sıfır olmayan yayılma süresi nedeniyle mükemmel çarpışmadan kaçınma ile sonuçlanmaz. Kablosuz ağlar CSMA'yı kullanabilir, ancak nadiren CD'yi kullanabilir, çünkü bir verici istasyonun yerel olarak güçlü sinyali, aynı zaman dilimi ve frekans bandında uzaktan iletilen yerel olarak zayıf bir sinyalin algılanmasını önler. Merkez istasyon modeli ile merkez istasyon, çoklu erişim kullanıcılarından farklı bir bantta yayın yapar. Çarpışmalar tespit edilir, ancak gidiş-dönüş yayılma süresi, iletimin durmasından yararlanmak için çok geç olabilir.

Çarpışma Çözünürlüğü Algoritmaları

Bu algoritmalar , bir çarpışmanın çözümünü zorlayarak oluklu ALOHA'nın verimliliğini artırır. Bir çarpışmadan sonra, çarpışma başarılarıyla çözülene kadar yalnızca çarpıştırıcıların (birikmiş küme) göndermesine izin verilir. Algoritma, çarpıştırıcıların minimum ortalama slot süresi sayısı içinde başarılı olmasını sağlarken, aynı zamanda tüm kullanıcıların çarpışmanın ne zaman çözüldüğünü bilmesine izin verir. Tüm potansiyel göndericilerin, önceki slotun bir başarı mı yoksa bir çarpışma mı yaşadığı veya boşta kalıp kalmadığı konusunda bilgilendirildiği varsayılır; Kaç tanesinin çarpıştığını bildikleri varsayılmıyor.

Rezervasyon ALOHA ve Hibrit Sistemler

Rezervasyon sistemlerinde, daha sonra gönderilecek verilerin rezervasyonunu yapmak için iletilen bilgiler, normalde toplam bilgi akışının çok küçük bir kısmıdır. Böylece kanalın küçük bir kısmı (alt kanal) rezervasyon trafiğine ayrılabilir ve hatta o alt kanal hafifçe kullanılabilir. ALOHA yöntemleri hafif bir trafik kanalında iyi çalışır, çünkü iletimler hızlıdır ve çarpışmalar nadirdir. Bir çarpışma yalnızca gecikmiş bir rezervasyon haline gelir ve bu nadir de olsa çok zahmetli değildir.

ALOHA ayrıca, bir TDM sistemindeki bir yuvayı, o yuvadaki verileri başarılı bir şekilde ileterek belirsiz bir ardışık çerçeve süresi için örtük olarak ayırmak için de kullanılabilir. Diğerleri, yuvanın boşta kaldığı gözlemlenene kadar bu yuvaya gönderme girişimini erteleyebilir. Bu yöntem *rezervasyon ALOHA* denir.

Slotted FAMA

Slotted FAMA (Floor Acquisition Multiple Access), özellikle **su altı akustik ağları** gibi yüksek gecikme sürelerine sahip ortamlarda kullanılan bir **MAC (Medium Access Control)** protokolüdür. Bu protokol, çarpışmaları azaltmayı ve ağ kaynaklarının daha verimli kullanılmasını hedefler. İşte Slotted FAMA'nın temel özellikleri ve örnek protokollerle karşılaştırılması:

Slotted FAMA'nın Temel Özellikleri

- Zaman Dilimleme (Slotted):** Veri iletimi, belirli zaman dilimlerine bölünerek gerçekleştirilir. Bu, çarpışma olasılığını azaltır.
- RTS/CTS Mekanizması:** Gönderici, veri göndermeden önce bir "RTS (Request to Send)" sinyali gönderir. Alıcı, "CTS (Clear to Send)" sinyaliyle yanıt verir. Bu mekanizma, çarpışmaları önlemeye yardımcı olur.
- Uyarlanabilir Geri Çekilme (Backoff):** Trafik yoğunluğuna bağlı olarak, gönderim için bekleme süreleri dinamik olarak ayarlanabilir.
- Uzun Gecikmelere Uyum:** Su altı ağlarında yaygın olan uzun gecikme sürelerine uygun şekilde tasarlanmıştır.

Slotted FAMA ve Diğer Protokoller

- ALOHA:** ALOHA, basit bir protokol olup çarpışmaları önlemek için herhangi bir mekanizma içermez. Slotted FAMA, ALOHA'ya göre daha karmaşık ancak daha verimlidir.

- **CSMA (Carrier Sense Multiple Access):** CSMA, kanalın boş olup olmadığını kontrol eder. Ancak, Slotted FAMA'nın RTS/CTS mekanizması, CSMA'ya göre daha iyi çarpışma önleme sağlar.
- **MACAW:** MACAW, CSMA protokolünün geliştirilmiş bir versiyonudur ve uyarlanabilir geri çekilme süreleri kullanır. Slotted FAMA, benzer bir yaklaşımı benimser ancak zaman dilimleme ile daha fazla kontrol sunar.

Slotted FAMA'nın Kullanım Alanları

- **Su Altı Akustik Ağlar:** Yüksek gecikme süreleri ve düşük bant genişliği nedeniyle bu protokol, su altı iletişimde oldukça etkilidir.
- **IoT (Nesnelerin İnterneti):** Özellikle enerji verimliliği ve çarpışma önleme gerektiren IoT uygulamalarında kullanılabilir.

Slotted FAMA, özellikle adil kaynak paylaşımı ve çarpışma önleme gereksinimlerinin kritik olduğu ağlarda tercih edilen bir protokoldür. Daha fazla detaylı inceleme için akademik makaleler veya teknik dokümanlar faydalı olabilir.

UW-MAC

UW-MAC protokolü, su altı kablosuz sensör ağları (UWSN) için özel olarak geliştirilmiş bir ortam erişim kontrol (MAC) protokolüdür. Bu protokol, geleneksel MAC protokollerinin su altı ortamındaki zorluklara yanıt verememesi nedeniyle tasarlanmıştır. İşte UW-MAC'in temel özellikleri ve incelenmesi gereken yönleri:

UW-MAC Protokolünün Temel Özellikleri

- **CDMA tabanlı erişim:** UW-MAC, çoklu erişim için CDMA (Code Division Multiple Access) kullanır. Bu, çoklu yol ve Doppler etkilerine karşı dayanıklılık sağlar.
- **Güç kontrollü erişim:** Hem verici hem de alıcı tabanlı CDMA kullanılarak enerji verimliliği artırılır.
- **Kümeleme yapısı:** Ağ, küme başları ve sensör düğümleri şeklinde organize edilir. Küme başları, TDMA zamanlaması ile ana istasyona veri iletir.
- **Gecikme ve veri aktarım hızı odaklı:** Geleneksel MAC protokollerinin aksine, UW-MAC enerji verimliliği yerine düşük gecikme ve yüksek veri aktarım hızını önceliklendirir.

Teknik Zorluklar ve UW-MAC'in Çözümleri

- **Uzun yayılım gecikmeleri:** Akustik iletişimde gecikmeler radyo frekanslarına göre çok daha uzundur. UW-MAC, bu gecikmeleri dikkate alarak zamanlama algoritmaları geliştirir.
- **Kısıtlı bant genişliği:** CDMA ile aynı frekansta birden fazla iletim mümkün hale getirilir.
- **Enerji sınırlamaları:** Güç kontrollü erişim sayesinde düğüm ömrü uzatılır.
- **Gizli ve açık terminal problemleri:** CDMA kodları sayesinde bu problemler minimize edilir.

Performans Değerlendirmesi

Simülasyonlar, UW-MAC'in aşağıdaki açılardan üstün olduğunu göstermiştir:

- **Daha düşük gecikme süresi**
- **Yüksek veri aktarım oranı**
- **Daha uzun ağ ömrü**