Delay cost model:

Sự sử dụng trung bình của trạm gốc là :

ρ(t) =∑x λ(x,t) / θ(x,t)

Tổng độ trễ của truy cập và truyền không giây là:

cwi(t) =∑x λ(x,t) / [θ(x,t)(1- ρ(t))]

Model độ trễ quá trình workload chịu tại server edge:

clo(t) là độ trễ xử lý bởi vì công suất tính toán hạn chế tại các server edge local.

Độ trễ truyền từ device edge đến local server là không đáng kể vì vị trí tương đồng về mặt vật lí. Để định lượng hiệu suất trễ của dịch vụ, như độ trễ trung bình và độ trễ đuôi (độ trễ 95%), mà không giới hạn mô hình vào bất kỳ hoạt động cụ thể nào, chúng tôi sử dụng khái niệm chung

clo (m(t),μ(t)) để thể hiện hiệu suất trễ trong khoảng thời gian t. Như q ví dụ cụa thể, chúng ta có thể mô hình quy trình dịch vụ tại 1 thể hiện server như 1 hàng đợi M/G/1 và sử dụng thời gian phản hồi trung bình, có thể được biểu diễn: clo(m(t),μ(t)) =u(t) / (m(t)·k−u(t)), với k là giá dịch vụ của mỗi server.

Với giảm tải công việc, chi phí coff(t) chủ yếu là độ trễ truyền do thời gian khứ hồi mạng(RTT), khác nhau tùy thuộc vào trạng thái tắc nghễn mạng. Mô hình đơn giản, độ trễ dịch vụ tại phía đám mây cũng bị hấp thụ vào trạng thái tắc nghẽn mạng. Do đó, chúng ta mô hình trạng thái tắc nghẽn mạng, kí hiệu bởi h(t), như 1 tham số ngoại sinh và biểu diễn nó trong mục RTT( cộng với độ trễ dịch vụ đám mây) cho đơn giản. RTT có thể được ước lượng từ sự đo đạc RTT trước (sử dụng chương trình ping) sử dụng trung bình di chuyển đơn giản hoặc tỉ mỉ hơn thuật toán chuỗi thời gian như phạm vi di chuyển tích hợp hoàn toàn(ARIMA). Độ trễ dịch vụ đám mây có thể được báo cáo bởi nhà cung cấp dịch vụ dựa trên tài nguyên tính toán có sẵn(CPU, bộ nhớ, …) cái mà nó có thể đưa ra. Do đó, chi phí độ trễ là

coff (h(t),λ(t),μ(t)) = (λ(t) – μ(t)) h(t)

Tổng chi phí độ trễ:

cdelay(h(t),λ(t),m(t),μ(t))=clo(m(t),μ(t)) +coff(h(t),λ(t),μ(t)) +cwi(λ(t))

C. Power model

Tổng năng lượng được đưa ra bởi các edge device trong khoảng thời gian bao gồm 2 phần:

Đầu tiên, hoạt động cơ bản và năng lượng truyền đưa ra bởi edge devices (các trạn cơ bản trong việc học) và thứ 2, năng lượng tính toán đưa ra bởi edge server. Phần đầu tiên nó phụ thuộc vào của chính sách giảm tải hoặc tự điều chỉnh, được mô hình hóa dop(λ(t)) = dsta + ddyn(λ(t)), trong đó, dsta là sự tiêu thụ năng lượng tĩnh và ddyn(λ(t)) là sự tiêu thụ năng lượng động phụ thuộc vào tổng khối lượng công việc. Năng lượng tính toán đưa ra phụ thuộc vào số server được bật như khối lượng công việc được xử lý cục bộ. Chúng tôi sử dụng hàm genneric dcom(m(t),μ(t)), cái mà đang tăng trong m(t) và μ(t), để chỉ rõ năng lượng tính toán được đưa ra. Tổng năng lượng đưa ra trong khoảng thời gian t là :

d(λ(t),m(t),μ(t)) =dop(λ(t)) +dcom(m(t),μ(t))

Để mô hình sự không chắc chắn của nguồn cung cấp năng lượng xanh, chúng tôi giả sử đó là ngân sách năng lương xanh, được thể hiện bởi g(t), được đóng băng thực hiện sau khi các quyết định giảm tải và tự điều chỉnh được đưa ra. Các quyết định không thể sử dụng thông tin chính xác của g(t). Tuy nhiên, giả định rằng có một trạng thái môi trường e(t) cái mà hệ thống có thể quan sát và nó mã hóa thông tin có giá trị của ngân sách năng lượng xanh được dự đoán trong khoảng thời gian hiện tại. Ví dụ, ban ngày tring thời tiết tốt ngụ ý ngân sách năng lương mặt trời cao nhất. Cụ thể, chúng ta mô hình g(t) như 1 biến ngẫu nhiên i.i.d từ e(t), mà tuân theo phân phối xác suất có điều kiện Pg(g(t)je(t)). Chú ý trạng thái môi trường e(t) có thể không là i.i.

D. Battery model

Batteries được sử dụng để cân bằng năng lương cung cấp và giảm tải. Trong 1 hệ thống mặt trời + gió, mô - đun quang điện và tua-bin gió có thể kết hợp đầu ra của chúng để cung cấp năng lượng cho hệ thống edge và sạc pin. Khi nỗ lực kết hợp là không đủ, pin sẽ thay thế để đảm bảo hoạt động ổn định của hệ thống. Chúng tôi chỉ rõ trạng thái pin tại thời điểm đầu của khoảng thời gian t bởi b(t) thuộc [0,B] = B (trong đơn vị của năng lượng) ở B là dung lượng pin. Vì lý do bảo vệ hệ thống, đơn vị pin phải được ngắt kết nối khỏi tải khi điện áp dưới mức dưới một ngưỡng nhất định để sạc. Chúng tôi map b(t) = 0 ngưỡng điện áp để đảm bảo hoạt động cơ bản của hệ thống.

Khi ngân sách năng lương xanh không thể dự đoán và do đó không biết tại thời điểm bắt đầu của khoảng thwoif gian t, hệ thống edge sử dụng 1 chính sách ước lượng thỏa mãn dcom(m(t),μ(t)) <= max{ b(t) – dop(λ(t)),0} để tránh kích hoạt năng lượng lưu trữ bằng cách đưa ra các quyết định giảm tải và tự điều chỉnh

Khi dop(λ(t)) >= b(t),dcom(λ(t),m(t),μ(t)) phải bằng 0, có nghĩa là hệ thống edge giảm tải tất cả công việc lên cloud nếu mức pin thậm chí không thể hỗ trợ hoạt đọng cơ bản và truyền trong khoảng thời gian hiện tại. nguồn điện dự phòng sẽ được sử dụng để duy trì hoạt động cơ bản cho khoảng tg. Chi phi do kich hoạt nguồn điện dự phòng là cbak(t) =φ\*dop(λ(t)) với φ > 0 là 1 hằng số thể hiện chi phí lớn do cung cấp nguồn điện dự phòng. Khoảng thời gian tiếp theo, trạng thái pin sau đó được mở rộng : b(t+ 1) =b(t) +g(t)  
Khi dop(λ(t)) <= b(t), hệ thống edge có thể xử lý phần công việc μ(t) <= λ(t) tại server, nhưng năng lượng đưa ra phải thỏa mãn dcom(λ(t),m(t),μ(t)) <= b(t) – dop(λ(t)). Tùy thuộc vào năng lượng xanh nhận được g(t) và năng lượng tính toán dcom(λ(t),m(t),μ(t)), pin sẽ được sạc lại hoặc xả tương ứng:

a) Nếu g(t) >= d(λ(t),m(t),μ(t)), độ dư g(t) – d(λ(t),m(t),μ(t)) được lưu trữ trong pin cho tới khi đạt được mức dung lượng B:

b(t+1) = max{b(t) + g(t) – d(λ(t),m(t),μ(t)) , B}

b) Nếu g(t) <= d(λ(t),m(t),μ(t)), sau đó pin sẽ được xả ra để bù đắp lượng thiếu hụt d(λ(t),m(t),μ(t)) -g(t):

b(t+1) = b(t) + g(t) – d(λ(t),m(t),μ(t))

Mô hình chi phí pin khử trong khoảng thời gian, kí hiệu bởi Cbattery(t), việc sử dụng lượng điện năng được xả trong khoảng thời gian này thường không được giối hạn. Cụ thể:

cbattery(t) =ω\*max { d(λ(t),m(t),μ(t)) – g(t),0}

ở đó , ω > 0 chi phí khấu hao đơn vị chuẩn hóa. Trong trương hợp thiếu hụt năng lượng, BS được cung cấp bởi năng lượng dự phòng như máy phát điện hoặc nhiên liệu năng lượng.

Tồn\_Tại e past(e), act(e, seeing), seer(e, John), seee(e, Marry), Lamda x Park(x) location(e, x)

ask(Tồn\_Tại e past(e), act(e, seeing), seer(e, John), seee(e, Marry), Lamda t location(e, t))