

模型解析의 理論과 應用(Ⅲ報)

—Queueing 模型—

朴 承 禹*

본 講座에서는 queueing 시스템 模型의 開發과 應用의 例를 中心으로 模擬發生 技法의 適用性에 대하여 考察하도록 한다. queueing 시스템이란 어떤 店舖에서 顧客이 訪問하여 用務를 마치는 狀況에서 待機하는 過程을 포함하는 경우를 말한다. 이와같은 시스템은 그 構成 因子들의 狀況이 時間에 따라 任意的으로 變化한다. 따라서, 特정한 入力資料에 대한 시스템의 反應은 狀況에 따라 다르며 이와같은 動的, 任意的 特性은 第Ⅰ, Ⅱ報의 物理的, 決定論的 接近方法和 區別되는 것이다.

Ⅳ. Queueing 시스템 模型化

가. 狀況 說明

queueing 시스템의 模型 開發과 應用을 紹介하기 위하여 비교적 單純한 1인 서비스 施設(single server system)에서의 顧客과 奉仕者의 狀況을 模擬發生하도록 한다. 이러한 서비스 施設로는 空港이나 버스 터미널 등의 1인 案内 窓이나 1인 이발소 등을 들 수 있겠다. (그림. 1 參照)

1인 서비스 施設에서는 한번에 한 顧客 만을 奉仕할 수 있으며 그 동안 다른 고객 등은 待機 狀態로 있게 된다. (그림. 1) 그러면 顧客의 平均 待機 時間은 얼마나 될 것이며 奉仕에 所要되는 時間은 얼마인가 하는 問題이다. 여기서 待機 時間은 顧客의 到着 時間으로 부터

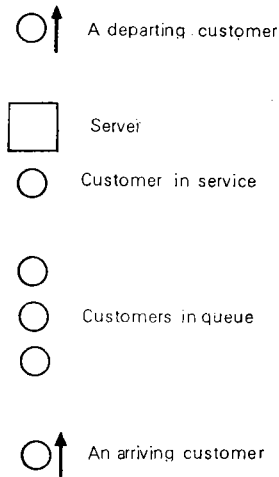


그림. 1. 1인 서비스 queueing 시스템의 例

봉사 받기 시작하는 시간까지도 定義할 수 있을 것이다.

顧客의 平均 待機 時間을 推定하기 위하여는 다음과 같은 狀況등이 模擬 發生되어야 할 것이다. 즉 ① 奉仕者의 狀況이 일하고 있는지 쉬고 있는지 與否와, ② queue(待機하는 줄이란 뜻임)에서의 顧客의 數, 그리고 ③ queue의 各 顧客의 到着 時間을 推定하는 것 등이다. 奉仕者들의 狀況은 顧客이 到着한 즉시 서비스를 받을 수 있는지 혹은 queue의 끝에서 待機하여야 하는 지를 決定하기 위하여 필요한 것이다. 한편 顧客의 서비스를 마친 다음 奉仕者는 queue의 고객이 있는 경우는 계속 일하는 狀況으로 될 것이며, queue의 고객의 數

*서울대학교 農科大學

가 零인 경우는 다음 顧客의 到着時間까지 쉬는 狀況이 될 것이다. 또한, 顧客의 到着時間은 그가 queue에서의 待機時間을 計算하는데 必要하다.

1인 서비스 施設에서는 2가지 事件이 있다. 顧客의 到着과 用務를 끝낸 顧客의 出發이 그것이다. 到着의 事件은 서비스 狀況을 쉬는 것에서 일하는 狀態로 하거나, 或은 queue의 顧客의 數가 1名 增加하게 되는 것이다. 出發의 事件은 이로 말미암아 到着의 事件과 反對現象이 發生되게 된다.

나. 時間의 前進機構

以上の 單純한 queue 問題는 動的이다. 즉 시스템의 特性이 時間에 따라 變化한다. 따라서 이 시스템의 模擬發生을 위하여는 模擬時間(simulation time)을 前進시켜야 할 것이다 이와같은 變數를 특히 模擬發生 時計, 혹은 模擬時計(simulation clock)라 한다.

模擬時計의 前進으로는 그 間隔을 일정하게 增加시켜가는 것과 다음 事件의 發生 時間을 기준하여 不規則적으로 增加하는 것이 可能하다. 前者를 一定間隔 時間前進法(fixed-increment time advance method) 그리고 後者를 次期事件 時間前進法(next-event time advance method)라고 한다. 여기서, 일정간격법은 차기사건법의 특별한 경우(즉 次期事件이 일정 간격으로 發生하는 條件일 때임)가 될 것이다. 따라서 次期事件法을 중심으로 考察하도록 한다.

먼저, 模擬時計를 零으로 하고, 未來의 事件의 發生 時間을 定義하도록 한다. 즉, 模擬時計를 맨 처음 事件이 發生되는 時點으로 前進시켜 그 時刻의 시스템의 狀況을 模擬發生하도록 하는 것이다. 그 다음 模擬時計를 다음 時間으로 前進시켜 必要한 狀況을 模擬發生하며 이를 반복하여 어떤 條件을 滿足시킬 때까지 時間을 變化하도록 한다. 따라서 시스템의 狀況은 事件 發生 時間에서만 變化가 可能하며, 시스템의 非活性 期間(inactivity period) 동안은 模擬時計가 도약하므로 模擬發生

에서 省略하게 되는 것이 이 方法의 特徵이 되겠다.

다. 次期事件時間

1인 서비스 queueing 시스템의 時間의 前進 方法의 例를 살펴보도록 한다. 그림. 2는 이러한 시스템의 時間에 따른 事件 發生 過程을 概略화한 것이다. 여기서,

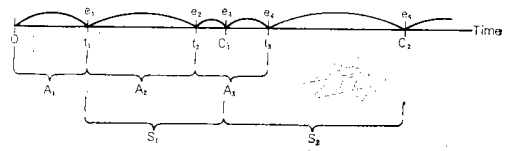


그림. 2. queueing 시스템의 次期事件時間 前進 方法의 例

t_i = i 번째 顧客의 到着時間

$A_i = t_i - t_{i-1} = i - 1$ 과 i 번째 到着時間 間隔

S_i = i 번째 顧客의 서비스 所要時間

D_i = i 번째 顧客의 queue의 待機時間

$C_i = t_i + D_i + S_i = i$ 번째 顧客이 서비스 종료 후 出發時間

e_i = i 번째의 事件發生時間이다. 한편, 以上の 各 變數는 任意的이다.

그림. 2의 次期事件時間의 模擬發生 方法은 다음과 같다. 우선 到着時間 間隔 A_1, A_2, \dots 와 서비스 所要時間 S_1, S_2, \dots 의 값이 特定 分布函數(통상적으로 確率分布函數를 適用할 수 있음) F_A 와 F_B 로서 定義할 수 있다고 假定하여 본다. 最初 $e_0 = 0$ 時間에는 奉仕者는 待機 狀態이다. 첫 顧客의 到着時間 t_1 을 F_A 函數로 부터 A_1 을 發生시켜 그 結果에 e_0 時間, 즉 零을 더한다. 이에 模擬時計를 e_0 로 부터 다음 事件의 發生時間인 $e_1 = t_1$ 으로 前進시킨다. 한편, t_1 時間에 到着한 顧客은 奉仕者가 待機狀態이므로 즉시 서비스를 받기 시작하게 될 것이며 그의 queue의 待機時間 $D_1 = 0$ 이 된다. 이 瞬間 奉仕者는 쉬는 상태로부터 일하는 狀態로 된다. 한편, 첫번째($i=1$)의 顧客의 서비스 종료시간 C_1 은 F_B 函數로 부터

S_1 을 發生시켜 그 결과를 t_1 시간에 더하면 된다.

두번째 ($i=2$) 顧客의 到着時間 t_2 는 A_2 를 F_A 函數로 부터 模擬發生한 結果를 $t_2=t_1+A_2$ 의 關係로 부터 定義할 수 있다. 이때 $t_2 < c_1$ 의 경우는 그림. 2에서와 같이 模擬時計를 e_1 으로부터 다음 事件 發生時間 $e_2=t_2$ 로 전진시킨다 (만약 $c_1 < t_2$ 이면 時計는 e_1 에서 c_1 으로 前進시켜야 한다). t_2 時刻에 到着한 顧客은 奉仕者가 일하는 상태에 있으므로 待機하여야 하며, 곧 queue의 顧客數가 1名으로 되고 그 到着時間이 記憶되나 서비스 時間 S_2 는 發生되지 않는다. 같은 方法으로 세번째의 到着時間 t_3 는 $t_3=t_2+A_3$ 로 定義하고, $c_1 < t_3$ 이면 模擬時計는 e_2 로 부터 다음 事件 發生時間 $e_3=c_1$ 으로 한다. 이 경우 用務를 마친 顧客은 出發하고, queue의 顧客(t_2 時刻에 도착한 者임)은 서비스를 받기 始作하며, 그의 queue에서의 待機時間과 서비스 완료시간을 각각 $D_2=c_1-t_2$ 와 $c_2=c_1+S_2$ 로 정의하며, queue의 顧客은 0名이 된다. 이와같은 方法으로 模擬發生을 進行하여 該當 變數가 特定한 값에 이르면 완료하도록 한다.

라. 電算 프로그램

以上에서는 單純한 queueing 시스템의 狀況이 模擬時計의 前進에 따라 變化하는 過程을 살펴 보았다. 이와같은 該當 시스템의 舉動을 具體적인 數值 등을 통하여 理解하는 것을 直觀的 說明(intuitive explanation)이라 한다. 直觀的 說明은 模型의 開發에서 매우 重要한 段階로서 그 結果는 電算 프로그램 作成에 應用된다.

模擬發生을 위한 電算 프로그램에는 여러 言語 등이 使用되고 있다. 그例는 FORTRAN ; GASP ; SLAM ; SIMULA 등이 있다. 여기서, FORTRAN은 일단 목적용 언어(general-purposed language)라고 하며, 其他의 특별 모의발생 언어와 區分한다. 여기서는 [FORTRAN 프로그램에 관하여 說明하도록 한다.

FORTRAN 프로그램의 作成을 위하여는

queueing 시스템의 直觀的 說明에서 考慮된 舉動은 물론 그밖의 시스템 特性에 대하여 보다 詳細하고 具體적인 假定 등을 마련하여야 한다. queueing 시스템의 경우 顧客의 到着時間과 奉仕時間의 적절한 模擬發生 方法 등의 檢討가 그것이다. 이러한 細部的 內容에서 問題에 관한 直觀的 理解와 실제 프로그램의 作成은 뚜렷히 區別되는 것이다.

한편, 顧客의 平均 待機時間을 決定하는 데 使用되는 顧客의 數, 즉 模擬發生 回數 또한 具體적인 數值나 基準 등으로서 表示하지 않으면 FORTRAN 프로그램이 不備하게 된다. 이때 模擬發生 回數는 結果에 影響을 미칠 수 있다. 例를 들어 顧客 10名의 待機時間으로부터 決定된 平均值와 100名의 結果, 혹은 1,000名의 경우가 각각 相異할 수 있는 까닭이다. 따라서 FORTRAN 프로그램에서는 시스템의 이와같은 特性에 관한 假定을 檢討하여 定義하도록 하는 것이 필요하다.

queueing 시스템에서 중요한 두가지의 變數는 앞서 言及한 바와 같이 顧客의 到着時間과 서비스 時間이다. 이들은 直觀的 說明에서 정의한 바와 같이 分布函數 F_A 와 F_s 로서 表示한 바 있다. FORTRAN 프로그램에서는 이들 函數를 정의하고 模擬發生을 통하여 고객의 도착시간과 고객별 서비스 시간을 發生하도록 하여야 한다. 여기서 分布函數 F_A F_s 는 실제 시스템의 資料를 적절한 分布 函數에 적용하여 정의 할 수 있겠다.

任意的 確率分布函數를 利用하여 無作爲의 變數를 發生하는 方法은 여러가지가 있겠으나 손 쉬운 方法의 하나를 소개하면 다음과 같다. 즉 0~1의 범위내에서 等分布를 보이는 無作爲의 變數를 얻고 (이러한 無作爲 變數의 發生方法 등은 電算機의 數學函數중의 한 種類이다). 그 結果를 分布函數에 代入하여 그 頻度を 정의 하므로써 推定值를 얻는 것이다.

마. 模擬發生 結果

queueing 시스템에서 顧客의 到着時間이 平

均 1分으로 그 分布가 指數函數分布 函數를 만족하며, 顧客別 서비스 時間은 平均 0.5分の 指數分布函數와 할 때, 顧客 1,000名에 대한 平均 待期時間의 模擬 發生의 結果는 0.497分 그리고 queue에서 待機중인 顧客數의 平均은 0.500名으로 各各 推定되었다. 이와 같은 模擬發生의 結果는 問題의 理論值인 0.500分과 0.500名에 매우 近似한 값이 있다.

한편, 模擬發生에서 얼마만큼의 반복 회수를 適用할 것인가는 용이한 문제가 아닐 수 있다. 앞서 顧客의 數를 1,000名으로 하였을 때 模擬發生의 結果와 理論值가 類似하였음을 考察하였으나 반드시 이러한 特定한 顧客의 數에 대한 平均値가 理論值에 接近하는 것은 아니다. 例로서 同一한 queueing 시스템의 경우 顧客의 到着速度를 平均 1分으로 부터 1.8分으로 80% 增加할 경우 平均 到着時間 間隔은 0.556分으로 하고 서비스 時間은 0.5分으로 같을 때 1,000名の 顧客의 平均 대기시간은 3.078分, 그리고 queue의 平均 고객은 5.501名으로 推定되었다. 이와 같은 結果는 理論值인 4.5分과 8.1名과 상당한 差異를 보여준다. 推定誤差는 소위 模擬發生의 靜止 時期를 決定하는 데에서 오는 것이다. 따라서, 적절한 靜止 方法의 選擇은 매우 重要하다.

다. 靜止 方法

模擬發生의 靜止 方法에는 여러가지 法則 등을 適用할 수 있다. 앞에서의 例와 같이 어떤 變數(즉 고객 수)가 特定한 값에 達할 때 靜止하는 方法이 있다. queueing 시스템의 경우 고객 수에 의한 靜止 方法에서는 靜止 時의 最終 模擬時計의 값은 任意의 값이 될 것이다. 그러나, 실제의 queueing 시스템에서는 時間이 任意의 값이 아닌 어떤 法則性을 갖는다. 예를 들면 總 時間이 8時間이 경과한 다음 靜止하는 것이다.

이와같이 靜止時間을 考察하는 것이 보다 實際的인 경우가 있다.

그러나, queueing 問題에서 到着時間(A_i 값)과 서비스 時間(S_i) 등의 값을 無作爲 變數로

假定한 경우 模擬時計가 8時間이 되는 時點에서 模擬發生이 靜止될 確率은 거의 0이 된다. 따라서 特定한 變數를 추가하여 종료 시각의 경과 후는 그 變數를 作動하도록 장치하여 그 變數가 發生되면 模擬發生을 靜止하는 方法을 생각할 수 있다. 이때는 queue에서의 고객 待機 수가 任意의 이 된다.

模擬時計를 午前 9時에서 午後 5時까지로 하여 5時가 경과한 다음은 顧客의 到着은 中止시키되 이 時間 以後에는 queue의 고객의 서비스를 완료할 때 까지 模擬發生을 持續하는 方法을 생각할 수도 있을 것이다. 실제로 銀行 등에서는 이와같이 運用을 하기도 한다.

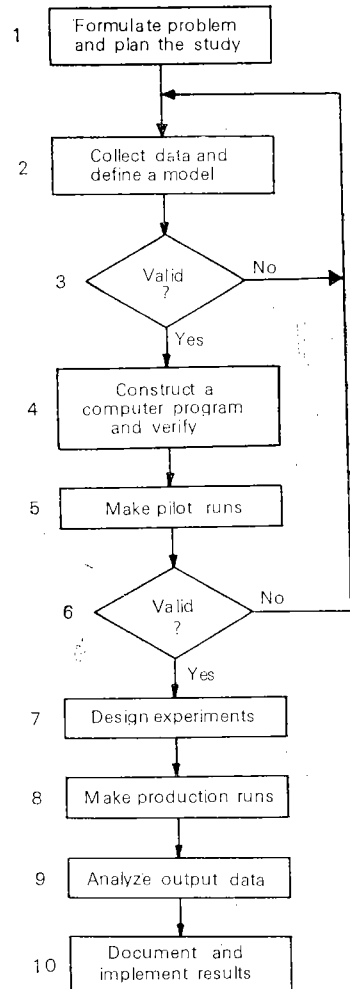


그림. 3. 模型研究의 推進段階

이와같은 靜止法則은 狀況에 따라 적절히 定義되고 또한 그 結果의 타당성과 基本 假定과의 關係등을 면밀히 검토하는 것이 必要하다.

사. 模型化의 段階

以上の queueing 시스템의 模型化 過程을 통하여 實際 問題의 模擬發生에 있어서 考慮되어야 할 몇 가지에 대하여 살펴 보았다. 여기서, 이와같은 시스템의 模擬發生의 段階를 整理하면 그림. 3에서와 같다.

즉,

- ① 問題의 設定,
- ② 資料의 蒐集과 模型의 定義,
- ③ 模型의 構成 要素의 檢討,
- ④ 電算 프로그램의 作成과 證明,
- ⑤ pilot run의 實施,
- ⑥ 模型의 妥當性 分析,

⑦ 實驗 計劃

⑧ 生産的 프로그램의 作成,

⑨ 出力資料의 分析, 그리고

⑩ 結果의 整理 등이다.

그림. 3의 模擬發生의 各 段階의 具體的 內容이나 作業量 등은 對象 시스템에 따라 變化할 수 있을 것이다. 筆者의 見解로는 뚜렷한 問題의 제기와 模型開發의 每 段階마다 그 成果를 면밀히 檢討하는 것이 특히 強調되어야 할 것이다. (끝)

《謝意》本 講座는 Law & Kelton(1982)의 "Simulation modeling and analysis" (McGraw-Hill Book Co. 出版)의 內容의 一部를 使用하였습니다. 또한 筆者는 본 講座가 持續되도록 바로 잡아 주시고 또한 內容의 補完과 보다 잘 理解될 수 있도록 配慮를 아끼지 않으신 農工學會誌 편집위원들께 謝意를 표합니다.